PRECIPITATION PROCESSING SYSTEM

GLOBAL PRECIPITATION MEASUREMENT

File Specification for GPM Products

Version 6.09 TKIO 3.94

January 17, 2020
Contents

1 Introduction ........................................... 41
   1.1 Identification ...................................... 41
   1.2 Scope ............................................... 41
   1.3 Purpose and Objectives .............................. 41
   1.4 Document Status and Schedule .................... 41
   1.5 Document Organization ............................. 41

2 Logical Format ........................................ 42
   2.1 swath Structure .................................... 42
   2.2 Grid Structure ...................................... 42

3 Physical Format ........................................ 42
   3.1 Hierarchical Data Format .......................... 42

4 Formatting Conventions ................................. 43
   4.1 File Structure Figure .............................. 43
   4.2 File Contents ...................................... 43
   4.3 Missing Data and Empty Granules ................ 43
   4.4 Array Dimension Order ............................. 44
   4.5 Array Index ........................................ 44
   4.6 Granule definition ................................ 44

5 Standard GPM Products ................................. 45
   5.1 1AGMI - GMI unpacked packet data ............... 45
   5.2 1ATMI - TMI unpacked packet data ............... 173
   5.3 1AVIRS - VIRS unpacked packet data .............. 246
   5.4 1BASETMI - TMI unpacked packet data .......... 301
   5.5 1BASEGMI - GMI Antenna Temperatures .......... 386
   5.6 1BASEGMIRSS - GMI Antenna Temperatures ....... 476
   5.7 1BASEGMIIXCAL - GMI Antenna Temperatures .... 566
   5.8 1BASESSMI - SSMI base ........................... 656
   5.9 1BASESSMIS - SSMIS base ........................ 679
   5.10 1BASEAMSRE - AMSRE base ...................... 704
5.11 1BASEAMSR2 - AMSR2 base ....................................... 718
5.12 1BASEWIND - Windsat base ........................................ 734
5.13 1BASEAMSUA - AMSUA base ........................................ 744
5.14 1BASEAMSUB - AMSUB base ........................................ 754
5.15 1BASEMHS - MHS base ............................................. 763
5.16 1BASESAPHIR - SAPHIR base ...................................... 772
5.17 1BASEATMS - ATMS base ........................................... 781
5.18 1BGMI - GMI Brightness Temperatures ............................ 795
5.19 1BTMI - TMI unpacked packet data ................................. 842
5.20 1BVIRS - VIRS Radiance ............................................ 909
5.21 1CTMI - GPM Common Calibrated Brightness Temperature ....... 928
5.22 1CGMI - GPM Common Calibrated Brightness Temperature ....... 949
5.23 1CSSMI - Common Calibrated Brightness Temperature ........... 964
5.24 1CSSMIS - Common Calibrated Brightness Temperature ........... 981
5.25 1CAMSRE - Common Calibrated Brightness Temperature .......... 1011
5.26 1CAMS2 - Common Calibrated Brightness Temperature .......... 1050
5.27 1CWM - Common Calibrated Brightness Temperature .............. 1089
5.28 1CMHS - Common Calibrated Brightness Temperature .............. 1099
5.29 1CSAPHIR - Common Calibrated Brightness Temperature ........... 1107
5.30 1CATMS - Common Calibrated Brightness Temperature ........... 1116
5.31 1CAMSUB - Common Calibrated Brightness Temperature .......... 1143
5.32 2AGPROFGMI - Radiometer Profiling ............................. 1151
5.33 2AGPROFTMI - Radiometer Profiling ............................. 1164
5.34 2AGPROFSSMI - Radiometer Profiling ............................ 1178
5.35 2AGPROFSSMIS - Radiometer Profiling ........................... 1191
5.36 2AGPROFAMSRE - Radiometer Profiling .......................... 1204
5.37 2AGPROFAMSR2 - Radiometer Profiling .......................... 1217
5.38 2AGPROFWIND - Radiometer Profiling ............................ 1230
5.39 2AGPROFAMSUB - Radiometer Profiling .......................... 1244
5.40 2AGPROFATMS - Radiometer Profiling ............................ 1257
5.41 2AGPROFMHS - Radiometer Profiling .............................. 1270
5.42 2APRPSSAPHIR - Radiometer Profiling ............................ 1283
5.43 3GPROF - GPROF Profiling ........................................ 1288
5.44 3PRPSSAPHIR - Gridded PRPS ........................................ 1294
5.45 1BKu - Ku Power ....................................................... 1296
5.46 1BKa - Ka Power ........................................................ 1328
5.47 1BPR - PR Power ........................................................ 1390
5.48 2AKu - Ku precipitation .................................................. 1421
5.49 2AKa - Ka precipitation .................................................. 1466
5.50 2ADPR - DPR precipitation .............................................. 1551
5.51 2APR - PR precipitation .................................................. 1686
5.52 3DPR - DPR Full Product .............................................. 1731
5.53 3DPRD - DPR Daily Product .......................................... 1830
5.54 3PR - PR Full Product ..................................................... 1841
5.55 3PRD - PR Daily Product .............................................. 1940
5.56 2BCMB - Level-2 DPR and GMI Combined ............................ 1951
5.57 3CMB - Combined precipitation ....................................... 2005
5.58 2BCMBT - Level-2 PR and TMI Combined ............................. 2030
5.59 3CMBT - Combined precipitation ...................................... 2058
5.60 2AKuX - Ku precipitation ............................................... 2083
5.61 2AKaX - Ka precipitation ............................................... 2129
5.62 2ADPRX - DPR precipitation ........................................... 2218
5.63 2AKuTMPX - Ku Temporary ............................................. 2314
5.64 2AKaTMPX - Ka Temporary ............................................. 2339
5.65 2ADPRTMP - DPR Temporary .......................................... 2387
5.66 2AKuENVX - Ku environment .......................................... 2418
5.67 2AKaENVX - Ka environment .......................................... 2424
5.68 2ADPRENVX - DPR environment ...................................... 2435
5.69 3DPRX - DPR Full Product ............................................. 2446
5.70 2BCMBX - Level-2 DPR and GMI Combined ............................ 2545
5.71 3CMBX - Combined precipitation ...................................... 2599
5.72 3CMBTX - Combined precipitation ..................................... 2624
5.73 3GSMAPH4 - GSMaP Hourly .......................................... 2648
5.74 3GSMAPM4 - GSMaP Monthly .......................................... 2653
5.75 3IMERGHH - IMERG 30-minute ...................................... 2656
5.76 3IMERGM - IMERG monthly ............................................ 2661
List of Figures

1 Data Format Structure for 1AGMI, GMI unpacked packet data ............. 47
2 Data Format Structure for 1AGMI, S1 ........................................ 48
3 Data Format Structure for 1AGMI, S2 ........................................ 49
4 Data Format Structure for 1AGMI, S3, S3 .................................... 50
5 Data Format Structure for 1AGMI, S3 ........................................ 51
6 Data Format Structure for 1AGMI, S4 ........................................ 51
7 Data Format Structure for 1AGMI, S5 ........................................ 51
8 Data Format Structure for 1AGMI, gmi1aHeader ................................ 51
9 Data Format Structure for 1AGMI, S1, ScanTime ............................ 52
10 Data Format Structure for 1AGMI, S1, scanStatus .......................... 52
11 Data Format Structure for 1AGMI, S1, navigation ........................... 53
12 Data Format Structure for 1AGMI, S1, sunData ............................. 53
13 Data Format Structure for 1AGMI, S2, ScanTime ............................ 54
14 Data Format Structure for 1AGMI, S2, scanStatus .......................... 54
15 Data Format Structure for 1AGMI, S2, navigation ........................... 55
16 Data Format Structure for 1AGMI, S2, sunData ............................. 55
17 Data Format Structure for 1AGMI, S3, ScanTime ............................ 56
18 Data Format Structure for 1AGMI, S3, TAM1 ................................ 56
19 Data Format Structure for 1AGMI, S3, TAM2 ................................ 56
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>Data Format Structure for 1AGMI, S3, TORQUE_BAR</td>
<td>56</td>
</tr>
<tr>
<td>21</td>
<td>Data Format Structure for 1AGMI, S3, GMI_TEMPERATURES</td>
<td>57</td>
</tr>
<tr>
<td>22</td>
<td>Data Format Structure for 1AGMI, S3, primaryHeader</td>
<td>57</td>
</tr>
<tr>
<td>23</td>
<td>Data Format Structure for 1AGMI, S3, GSDR_TIME</td>
<td>57</td>
</tr>
<tr>
<td>24</td>
<td>Data Format Structure for 1AGMI, SENSOR_INFO</td>
<td>58</td>
</tr>
<tr>
<td>25</td>
<td>Data Format Structure for 1AGMI, S3, SENSOR_INFO</td>
<td>59</td>
</tr>
<tr>
<td>26</td>
<td>Data Format Structure for 1AGMI, RS_INFO</td>
<td>60</td>
</tr>
<tr>
<td>27</td>
<td>Data Format Structure for 1AGMI, S3, RS_INFO</td>
<td>61</td>
</tr>
<tr>
<td>28</td>
<td>Data Format Structure for 1AGMI, SYNCH_STAMPS</td>
<td>62</td>
</tr>
<tr>
<td>29</td>
<td>Data Format Structure for 1AGMI, S3, SYNCH_STAMPS</td>
<td>63</td>
</tr>
<tr>
<td>30</td>
<td>Data Format Structure for 1AGMI, SYNCH_STAMPS2</td>
<td>64</td>
</tr>
<tr>
<td>31</td>
<td>Data Format Structure for 1AGMI, S3, SYNCH_STAMPS2</td>
<td>65</td>
</tr>
<tr>
<td>32</td>
<td>Data Format Structure for 1AGMI, RSHSK_STATUS</td>
<td>66</td>
</tr>
<tr>
<td>33</td>
<td>Data Format Structure for 1AGMI, S3, RSHSK_STATUS</td>
<td>67</td>
</tr>
<tr>
<td>34</td>
<td>Data Format Structure for 1AGMI, S3, RSHSK_SAMPL_INFO</td>
<td>67</td>
</tr>
<tr>
<td>35</td>
<td>Data Format Structure for 1AGMI, S3, RSHSK_GAIN</td>
<td>68</td>
</tr>
<tr>
<td>36</td>
<td>Data Format Structure for 1AGMI, RSHSK_TEMP</td>
<td>69</td>
</tr>
<tr>
<td>37</td>
<td>Data Format Structure for 1AGMI, S3, RSHSK_TEMP</td>
<td>70</td>
</tr>
<tr>
<td>38</td>
<td>Data Format Structure for 1AGMI, IEHSK_TEMP</td>
<td>71</td>
</tr>
<tr>
<td>39</td>
<td>Data Format Structure for 1AGMI, S3, IEHSK_TEMP</td>
<td>72</td>
</tr>
<tr>
<td>40</td>
<td>Data Format Structure for 1AGMI, IE_TELEMETRY</td>
<td>73</td>
</tr>
<tr>
<td>41</td>
<td>Data Format Structure for 1AGMI, S3, IE_TELEMETRY</td>
<td>74</td>
</tr>
<tr>
<td>42</td>
<td>Data Format Structure for 1AGMI, MECHANISMS</td>
<td>75</td>
</tr>
<tr>
<td>43</td>
<td>Data Format Structure for 1AGMI, S3, MECHANISMS</td>
<td>76</td>
</tr>
<tr>
<td>44</td>
<td>Data Format Structure for 1AGMI, SMPL_INFO</td>
<td>77</td>
</tr>
<tr>
<td>45</td>
<td>Data Format Structure for 1AGMI, S3, SMPL_INFO</td>
<td>78</td>
</tr>
<tr>
<td>46</td>
<td>Data Format Structure for 1AGMI, S4, ScanTime</td>
<td>79</td>
</tr>
<tr>
<td>47</td>
<td>Data Format Structure for 1AGMI, S5, ScanTime</td>
<td>80</td>
</tr>
<tr>
<td>48</td>
<td>Data Format Structure for 1ATMI, TMI unpacked packet data</td>
<td>176</td>
</tr>
<tr>
<td>49</td>
<td>Data Format Structure for 1ATMI, S1</td>
<td>177</td>
</tr>
<tr>
<td>50</td>
<td>Data Format Structure for 1ATMI, S2</td>
<td>178</td>
</tr>
<tr>
<td>51</td>
<td>Data Format Structure for 1ATMI, S3</td>
<td>179</td>
</tr>
<tr>
<td>52</td>
<td>Data Format Structure for 1ATMI, S4</td>
<td>180</td>
</tr>
</tbody>
</table>
### LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>Data Format Structure for 1ATMI, tmi1aHeader</td>
<td>180</td>
</tr>
<tr>
<td>54</td>
<td>Data Format Structure for 1ATMI, S1, ScanTime</td>
<td>180</td>
</tr>
<tr>
<td>55</td>
<td>Data Format Structure for 1ATMI, S1, scanStatus</td>
<td>181</td>
</tr>
<tr>
<td>56</td>
<td>Data Format Structure for 1ATMI, S1, primaryHeader</td>
<td>181</td>
</tr>
<tr>
<td>57</td>
<td>Data Format Structure for 1ATMI, S1, navigation</td>
<td>182</td>
</tr>
<tr>
<td>58</td>
<td>Data Format Structure for 1ATMI, S1, sunData</td>
<td>182</td>
</tr>
<tr>
<td>59</td>
<td>Data Format Structure for 1ATMI, S2, ScanTime</td>
<td>183</td>
</tr>
<tr>
<td>60</td>
<td>Data Format Structure for 1ATMI, S2, scanStatus</td>
<td>183</td>
</tr>
<tr>
<td>61</td>
<td>Data Format Structure for 1ATMI, S2, primaryHeader</td>
<td>184</td>
</tr>
<tr>
<td>62</td>
<td>Data Format Structure for 1ATMI, S2, navigation</td>
<td>185</td>
</tr>
<tr>
<td>63</td>
<td>Data Format Structure for 1ATMI, S2, sunData</td>
<td>186</td>
</tr>
<tr>
<td>64</td>
<td>Data Format Structure for 1ATMI, S3, ScanTime</td>
<td>186</td>
</tr>
<tr>
<td>65</td>
<td>Data Format Structure for 1ATMI, S3, scanStatus</td>
<td>187</td>
</tr>
<tr>
<td>66</td>
<td>Data Format Structure for 1ATMI, S3, primaryHeader</td>
<td>187</td>
</tr>
<tr>
<td>67</td>
<td>Data Format Structure for 1ATMI, S3, navigation</td>
<td>188</td>
</tr>
<tr>
<td>68</td>
<td>Data Format Structure for 1ATMI, S3, sunData</td>
<td>188</td>
</tr>
<tr>
<td>69</td>
<td>Data Format Structure for 1ATMI, S4, ScanTime</td>
<td>189</td>
</tr>
<tr>
<td>70</td>
<td>Data Format Structure for 1ATMI, S4, primaryHeader</td>
<td>189</td>
</tr>
<tr>
<td>71</td>
<td>Data Format Structure for 1ATMI, TMIHKPACKET</td>
<td>190</td>
</tr>
<tr>
<td>72</td>
<td>Data Format Structure for 1ATMI, S4, TMIHKPACKET</td>
<td>191</td>
</tr>
<tr>
<td>73</td>
<td>Data Format Structure for 1AVIRS, VIRS unpacked packet data</td>
<td>248</td>
</tr>
<tr>
<td>74</td>
<td>Data Format Structure for 1AVIRS, S1</td>
<td>249</td>
</tr>
<tr>
<td>75</td>
<td>Data Format Structure for 1AVIRS, S2</td>
<td>250</td>
</tr>
<tr>
<td>76</td>
<td>Data Format Structure for 1AVIRS, S1, ScanTime</td>
<td>250</td>
</tr>
<tr>
<td>77</td>
<td>Data Format Structure for 1AVIRS, S1, scanStatus</td>
<td>251</td>
</tr>
<tr>
<td>78</td>
<td>Data Format Structure for 1AVIRS, S1, primaryHeader</td>
<td>251</td>
</tr>
<tr>
<td>79</td>
<td>Data Format Structure for 1AVIRS, S1, navigation</td>
<td>252</td>
</tr>
<tr>
<td>80</td>
<td>Data Format Structure for 1AVIRS, S1, solarCal</td>
<td>252</td>
</tr>
<tr>
<td>81</td>
<td>Data Format Structure for 1AVIRS, S1, sunData</td>
<td>252</td>
</tr>
<tr>
<td>82</td>
<td>Data Format Structure for 1AVIRS, virsPacketHeader</td>
<td>253</td>
</tr>
<tr>
<td>83</td>
<td>Data Format Structure for 1AVIRS, S1, virsPacketHeader</td>
<td>254</td>
</tr>
<tr>
<td>84</td>
<td>Data Format Structure for 1AVIRS, S2, ScanTime</td>
<td>255</td>
</tr>
<tr>
<td>85</td>
<td>Data Format Structure for 1AVIRS, S2, primaryHeader</td>
<td>255</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>86</td>
<td>Data Format Structure for 1AVIRS, virsPacketHeader</td>
<td>257</td>
</tr>
<tr>
<td>87</td>
<td>Data Format Structure for 1AVIRS, S2, virsPacketHeader</td>
<td>258</td>
</tr>
<tr>
<td>88</td>
<td>Data Format Structure for 1AVIRS, VIRSHKPACKET</td>
<td>259</td>
</tr>
<tr>
<td>89</td>
<td>Data Format Structure for 1AVIRS, S2, VIRSHKPACKET</td>
<td>260</td>
</tr>
<tr>
<td>90</td>
<td>Data Format Structure for 1BASETMI, TMI unpacked packet data</td>
<td>304</td>
</tr>
<tr>
<td>91</td>
<td>Data Format Structure for 1BASETMI, S1</td>
<td>305</td>
</tr>
<tr>
<td>92</td>
<td>Data Format Structure for 1BASETMI, S2</td>
<td>306</td>
</tr>
<tr>
<td>93</td>
<td>Data Format Structure for 1BASETMI, S3</td>
<td>307</td>
</tr>
<tr>
<td>94</td>
<td>Data Format Structure for 1BASETMI, S4</td>
<td>308</td>
</tr>
<tr>
<td>95</td>
<td>Data Format Structure for 1BASETMI, S1, ScanTime</td>
<td>308</td>
</tr>
<tr>
<td>96</td>
<td>Data Format Structure for 1BASETMI, S1, scanStatus</td>
<td>309</td>
</tr>
<tr>
<td>97</td>
<td>Data Format Structure for 1BASETMI, S1, navigation</td>
<td>310</td>
</tr>
<tr>
<td>98</td>
<td>Data Format Structure for 1BASETMI, S1, calibration</td>
<td>310</td>
</tr>
<tr>
<td>99</td>
<td>Data Format Structure for 1BASETMI, S1, cal2</td>
<td>311</td>
</tr>
<tr>
<td>100</td>
<td>Data Format Structure for 1BASETMI, S1, calCounts</td>
<td>311</td>
</tr>
<tr>
<td>101</td>
<td>Data Format Structure for 1BASETMI, S1, sunData</td>
<td>311</td>
</tr>
<tr>
<td>102</td>
<td>Data Format Structure for 1BASETMI, S2, ScanTime</td>
<td>311</td>
</tr>
<tr>
<td>103</td>
<td>Data Format Structure for 1BASETMI, S2, scanStatus</td>
<td>312</td>
</tr>
<tr>
<td>104</td>
<td>Data Format Structure for 1BASETMI, S2, navigation</td>
<td>313</td>
</tr>
<tr>
<td>105</td>
<td>Data Format Structure for 1BASETMI, S2, calibration</td>
<td>313</td>
</tr>
<tr>
<td>106</td>
<td>Data Format Structure for 1BASETMI, S2, cal2</td>
<td>314</td>
</tr>
<tr>
<td>107</td>
<td>Data Format Structure for 1BASETMI, S2, calCounts</td>
<td>314</td>
</tr>
<tr>
<td>108</td>
<td>Data Format Structure for 1BASETMI, S2, sunData</td>
<td>314</td>
</tr>
<tr>
<td>109</td>
<td>Data Format Structure for 1BASETMI, S3, ScanTime</td>
<td>314</td>
</tr>
<tr>
<td>110</td>
<td>Data Format Structure for 1BASETMI, S3, scanStatus</td>
<td>315</td>
</tr>
<tr>
<td>111</td>
<td>Data Format Structure for 1BASETMI, S3, navigation</td>
<td>315</td>
</tr>
<tr>
<td>112</td>
<td>Data Format Structure for 1BASETMI, S3, calibration</td>
<td>315</td>
</tr>
<tr>
<td>113</td>
<td>Data Format Structure for 1BASETMI, S3, cal2</td>
<td>317</td>
</tr>
<tr>
<td>114</td>
<td>Data Format Structure for 1BASETMI, S3, calCounts</td>
<td>318</td>
</tr>
<tr>
<td>115</td>
<td>Data Format Structure for 1BASETMI, S3, sunData</td>
<td>318</td>
</tr>
<tr>
<td>116</td>
<td>Data Format Structure for 1BASETMI, S4, ScanTime</td>
<td>319</td>
</tr>
<tr>
<td>117</td>
<td>Data Format Structure for 1BASETMI, S4, primaryHeader</td>
<td>319</td>
</tr>
<tr>
<td>118</td>
<td>Data Format Structure for 1BASETMI, TMIHKPACKET</td>
<td>320</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

119 Data Format Structure for 1BASETMI, S4, TMIHKPACKET ........... 321
120 Data Format Structure for 1BASEGMI, GMI Antenna Temperatures ... 387
121 Data Format Structure for 1BASEGMI, S1 .................................. 388
122 Data Format Structure for 1BASEGMI, S2 ............................... 389
123 Data Format Structure for 1BASEGMI, S3 ............................... 390
124 Data Format Structure for 1BASEGMI, S4 ............................... 390
125 Data Format Structure for 1BASEGMI, S1, ScanTime ................. 390
126 Data Format Structure for 1BASEGMI, S1, scanStatus ............... 391
127 Data Format Structure for 1BASEGMI, S1, sampleHeader ............. 391
128 Data Format Structure for 1BASEGMI, S1, NEDTinfo ................ 391
129 Data Format Structure for 1BASEGMI, S1, navigation ................ 392
130 Data Format Structure for 1BASEGMI, S1, nav2 ....................... 392
131 Data Format Structure for 1BASEGMI, S1, calibration ................ 393
132 Data Format Structure for 1BASEGMI, cal2 .............................. 394
133 Data Format Structure for 1BASEGMI, S1, cal2 ......................... 395
134 Data Format Structure for 1BASEGMI, S1, calCounts ................ 395
135 Data Format Structure for 1BASEGMI, S1, sunData ................... 396
136 Data Format Structure for 1BASEGMI, S2, ScanTime .................... 396
137 Data Format Structure for 1BASEGMI, S2, scanStatus ................ 397
138 Data Format Structure for 1BASEGMI, S2, sampleHeader .............. 397
139 Data Format Structure for 1BASEGMI, S2, NEDTinfo .................. 397
140 Data Format Structure for 1BASEGMI, S2, navigation ................ 398
141 Data Format Structure for 1BASEGMI, S2, nav2 ....................... 398
142 Data Format Structure for 1BASEGMI, S2, calibration ................ 399
143 Data Format Structure for 1BASEGMI, cal2 .............................. 400
144 Data Format Structure for 1BASEGMI, S2, cal2 ......................... 401
145 Data Format Structure for 1BASEGMI, S2, calCounts ................ 401
146 Data Format Structure for 1BASEGMI, S2, sunData ................... 402
147 Data Format Structure for 1BASEGMI, S3, ScanTime .................... 402
148 Data Format Structure for 1BASEGMI, S3, scanStatus ................ 403
149 Data Format Structure for 1BASEGMI, S3, calibration ................ 403
150 Data Format Structure for 1BASEGMI, S4, ScanTime .................... 404
151 Data Format Structure for 1BASEGMI, S4, scanStatus ................ 404
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>152</td>
<td>Data Format Structure for 1BASEGMI, S4, calibration</td>
<td>405</td>
</tr>
<tr>
<td>153</td>
<td>Data Format Structure for 1BASEGMI, GMI Antenna Temperatures</td>
<td>477</td>
</tr>
<tr>
<td>154</td>
<td>Data Format Structure for 1BASEGMI, S1</td>
<td>478</td>
</tr>
<tr>
<td>155</td>
<td>Data Format Structure for 1BASEGMI, S2</td>
<td>479</td>
</tr>
<tr>
<td>156</td>
<td>Data Format Structure for 1BASEGMI, S3</td>
<td>480</td>
</tr>
<tr>
<td>157</td>
<td>Data Format Structure for 1BASEGMI, S4</td>
<td>480</td>
</tr>
<tr>
<td>158</td>
<td>Data Format Structure for 1BASEGMI, S1, ScanTime</td>
<td>480</td>
</tr>
<tr>
<td>159</td>
<td>Data Format Structure for 1BASEGMI, S1, scanStatus</td>
<td>481</td>
</tr>
<tr>
<td>160</td>
<td>Data Format Structure for 1BASEGMI, S1, sampleHeader</td>
<td>481</td>
</tr>
<tr>
<td>161</td>
<td>Data Format Structure for 1BASEGMI, S1, NEDTinfo</td>
<td>481</td>
</tr>
<tr>
<td>162</td>
<td>Data Format Structure for 1BASEGMI, S1, navigation</td>
<td>482</td>
</tr>
<tr>
<td>163</td>
<td>Data Format Structure for 1BASEGMI, S1, nav2</td>
<td>482</td>
</tr>
<tr>
<td>164</td>
<td>Data Format Structure for 1BASEGMI, S1, calibration</td>
<td>483</td>
</tr>
<tr>
<td>165</td>
<td>Data Format Structure for 1BASEGMI, cal2</td>
<td>484</td>
</tr>
<tr>
<td>166</td>
<td>Data Format Structure for 1BASEGMI, S1, cal2</td>
<td>485</td>
</tr>
<tr>
<td>167</td>
<td>Data Format Structure for 1BASEGMI, S1, calCounts</td>
<td>485</td>
</tr>
<tr>
<td>168</td>
<td>Data Format Structure for 1BASEGMI, S1, sunData</td>
<td>486</td>
</tr>
<tr>
<td>169</td>
<td>Data Format Structure for 1BASEGMI, S2, ScanTime</td>
<td>486</td>
</tr>
<tr>
<td>170</td>
<td>Data Format Structure for 1BASEGMI, S2, scanStatus</td>
<td>487</td>
</tr>
<tr>
<td>171</td>
<td>Data Format Structure for 1BASEGMI, S2, sampleHeader</td>
<td>487</td>
</tr>
<tr>
<td>172</td>
<td>Data Format Structure for 1BASEGMI, S2, NEDTinfo</td>
<td>487</td>
</tr>
<tr>
<td>173</td>
<td>Data Format Structure for 1BASEGMI, S2, navigation</td>
<td>488</td>
</tr>
<tr>
<td>174</td>
<td>Data Format Structure for 1BASEGMI, S2, nav2</td>
<td>488</td>
</tr>
<tr>
<td>175</td>
<td>Data Format Structure for 1BASEGMI, S2, calibration</td>
<td>489</td>
</tr>
<tr>
<td>176</td>
<td>Data Format Structure for 1BASEGMI, cal2</td>
<td>490</td>
</tr>
<tr>
<td>177</td>
<td>Data Format Structure for 1BASEGMI, S2, cal2</td>
<td>491</td>
</tr>
<tr>
<td>178</td>
<td>Data Format Structure for 1BASEGMI, S2, calCounts</td>
<td>491</td>
</tr>
<tr>
<td>179</td>
<td>Data Format Structure for 1BASEGMI, S2, sunData</td>
<td>492</td>
</tr>
<tr>
<td>180</td>
<td>Data Format Structure for 1BASEGMI, S3, ScanTime</td>
<td>492</td>
</tr>
<tr>
<td>181</td>
<td>Data Format Structure for 1BASEGMI, S3, scanStatus</td>
<td>493</td>
</tr>
<tr>
<td>182</td>
<td>Data Format Structure for 1BASEGMI, S3, calibration</td>
<td>493</td>
</tr>
<tr>
<td>183</td>
<td>Data Format Structure for 1BASEGMI, S4, ScanTime</td>
<td>494</td>
</tr>
<tr>
<td>184</td>
<td>Data Format Structure for 1BASEGMI, S4, scanStatus</td>
<td>494</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>185</td>
<td>Data Format Structure for 1BASEGMIRSS, S4, calibration</td>
<td></td>
</tr>
<tr>
<td>186</td>
<td>Data Format Structure for 1BASEGMIXCAL, GMI Antenna Temperatures</td>
<td></td>
</tr>
<tr>
<td>187</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1</td>
<td></td>
</tr>
<tr>
<td>188</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2</td>
<td></td>
</tr>
<tr>
<td>189</td>
<td>Data Format Structure for 1BASEGMIXCAL, S3</td>
<td></td>
</tr>
<tr>
<td>190</td>
<td>Data Format Structure for 1BASEGMIXCAL, S4</td>
<td></td>
</tr>
<tr>
<td>191</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, ScanTime</td>
<td></td>
</tr>
<tr>
<td>192</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, scanStatus</td>
<td></td>
</tr>
<tr>
<td>193</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, sampleHeader</td>
<td></td>
</tr>
<tr>
<td>194</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, NEDTinfo</td>
<td></td>
</tr>
<tr>
<td>195</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, navigation</td>
<td></td>
</tr>
<tr>
<td>196</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, nav2</td>
<td></td>
</tr>
<tr>
<td>197</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, calibration</td>
<td></td>
</tr>
<tr>
<td>198</td>
<td>Data Format Structure for 1BASEGMIXCAL, cal2</td>
<td></td>
</tr>
<tr>
<td>199</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, cal2</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, calCounts</td>
<td></td>
</tr>
<tr>
<td>201</td>
<td>Data Format Structure for 1BASEGMIXCAL, S1, sunData</td>
<td></td>
</tr>
<tr>
<td>202</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, ScanTime</td>
<td></td>
</tr>
<tr>
<td>203</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, scanStatus</td>
<td></td>
</tr>
<tr>
<td>204</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, sampleHeader</td>
<td></td>
</tr>
<tr>
<td>205</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, NEDTinfo</td>
<td></td>
</tr>
<tr>
<td>206</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, navigation</td>
<td></td>
</tr>
<tr>
<td>207</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, nav2</td>
<td></td>
</tr>
<tr>
<td>208</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, calibration</td>
<td></td>
</tr>
<tr>
<td>209</td>
<td>Data Format Structure for 1BASEGMIXCAL, cal2</td>
<td></td>
</tr>
<tr>
<td>210</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, cal2</td>
<td></td>
</tr>
<tr>
<td>211</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, calCounts</td>
<td></td>
</tr>
<tr>
<td>212</td>
<td>Data Format Structure for 1BASEGMIXCAL, S2, sunData</td>
<td></td>
</tr>
<tr>
<td>213</td>
<td>Data Format Structure for 1BASEGMIXCAL, S3, ScanTime</td>
<td></td>
</tr>
<tr>
<td>214</td>
<td>Data Format Structure for 1BASEGMIXCAL, S3, scanStatus</td>
<td></td>
</tr>
<tr>
<td>215</td>
<td>Data Format Structure for 1BASEGMIXCAL, S3, calibration</td>
<td></td>
</tr>
<tr>
<td>216</td>
<td>Data Format Structure for 1BASEGMIXCAL, S4, ScanTime</td>
<td></td>
</tr>
<tr>
<td>217</td>
<td>Data Format Structure for 1BASEGMIXCAL, S4, scanStatus</td>
<td></td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>218</td>
<td>Data Format Structure for 1BASEGMIXCAL, S4, calibration</td>
<td>585</td>
</tr>
<tr>
<td>219</td>
<td>Data Format Structure for 1BASESSMI, SSMI base</td>
<td>657</td>
</tr>
<tr>
<td>220</td>
<td>Data Format Structure for 1BASESSMI, SSMI base</td>
<td>658</td>
</tr>
<tr>
<td>221</td>
<td>Data Format Structure for 1BASESSMI, SSMI base</td>
<td>659</td>
</tr>
<tr>
<td>222</td>
<td>Data Format Structure for 1BASESSMI, SSMI base</td>
<td>660</td>
</tr>
<tr>
<td>223</td>
<td>Data Format Structure for 1BASESSMI, SSMI base</td>
<td>661</td>
</tr>
<tr>
<td>224</td>
<td>Data Format Structure for 1BASESSMI, SSMI base</td>
<td>662</td>
</tr>
<tr>
<td>225</td>
<td>Data Format Structure for 1BASESSMI, baseHeader</td>
<td>662</td>
</tr>
<tr>
<td>226</td>
<td>Data Format Structure for 1BASESSMI, ScanTime</td>
<td>663</td>
</tr>
<tr>
<td>227</td>
<td>Data Format Structure for 1BASESSMIS, SSMIS base</td>
<td>680</td>
</tr>
<tr>
<td>228</td>
<td>Data Format Structure for 1BASESSMIS, SSMIS base</td>
<td>681</td>
</tr>
<tr>
<td>229</td>
<td>Data Format Structure for 1BASESSMIS, SSMIS base</td>
<td>682</td>
</tr>
<tr>
<td>230</td>
<td>Data Format Structure for 1BASESSMIS, SSMIS base</td>
<td>683</td>
</tr>
<tr>
<td>231</td>
<td>Data Format Structure for 1BASESSMIS, SSMIS base</td>
<td>684</td>
</tr>
<tr>
<td>232</td>
<td>Data Format Structure for 1BASESSMIS, SSMIS base</td>
<td>685</td>
</tr>
<tr>
<td>233</td>
<td>Data Format Structure for 1BASESSMIS, baseHeader</td>
<td>686</td>
</tr>
<tr>
<td>234</td>
<td>Data Format Structure for 1BASESSMIS, ScanTime</td>
<td>686</td>
</tr>
<tr>
<td>235</td>
<td>Data Format Structure for 1BASEAMSRE, AMSRE base</td>
<td>706</td>
</tr>
<tr>
<td>236</td>
<td>Data Format Structure for 1BASEAMSRE, AMSRE base</td>
<td>707</td>
</tr>
<tr>
<td>237</td>
<td>Data Format Structure for 1BASEAMSRE, AMSRE base</td>
<td>708</td>
</tr>
<tr>
<td>238</td>
<td>Data Format Structure for 1BASEAMSRE, baseHeader</td>
<td>708</td>
</tr>
<tr>
<td>239</td>
<td>Data Format Structure for 1BASEAMSRE, ScanTime</td>
<td>709</td>
</tr>
<tr>
<td>240</td>
<td>Data Format Structure for 1BASEAMSR2, AMSR2 base</td>
<td>720</td>
</tr>
<tr>
<td>241</td>
<td>Data Format Structure for 1BASEAMSR2, AMSR2 base</td>
<td>721</td>
</tr>
<tr>
<td>242</td>
<td>Data Format Structure for 1BASEAMSR2, AMSR2 base</td>
<td>722</td>
</tr>
<tr>
<td>243</td>
<td>Data Format Structure for 1BASEAMSR2, AMSR2 base</td>
<td>723</td>
</tr>
<tr>
<td>244</td>
<td>Data Format Structure for 1BASEAMSR2, baseHeader</td>
<td>723</td>
</tr>
<tr>
<td>245</td>
<td>Data Format Structure for 1BASEAMSR2, ScanTime</td>
<td>723</td>
</tr>
<tr>
<td>246</td>
<td>Data Format Structure for 1BASEWIND, Windsat base</td>
<td>735</td>
</tr>
<tr>
<td>247</td>
<td>Data Format Structure for 1BASEWIND, Windsat base</td>
<td>736</td>
</tr>
<tr>
<td>248</td>
<td>Data Format Structure for 1BASEWIND, baseHeader</td>
<td>736</td>
</tr>
<tr>
<td>249</td>
<td>Data Format Structure for 1BASEWIND, ScanTime</td>
<td>736</td>
</tr>
<tr>
<td>250</td>
<td>Data Format Structure for 1BASEAMSUA, AMSUA base</td>
<td>745</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

251 Data Format Structure for 1BASEAMSUA, AMSUA base .......................... 746
252 Data Format Structure for 1BASEAMSUA, baseHeader .......................... 746
253 Data Format Structure for 1BASEAMSUA, ScanTime ............................ 747
254 Data Format Structure for 1BASEAMSUB, AMSUB base .......................... 755
255 Data Format Structure for 1BASEAMSUB, AMSUB base .......................... 756
256 Data Format Structure for 1BASEAMSUB, baseHeader .......................... 756
257 Data Format Structure for 1BASEAMSUB, ScanTime ............................ 756
258 Data Format Structure for 1BASEMHS, MHS base .................................. 764
259 Data Format Structure for 1BASEMHS, MHS base .................................. 765
260 Data Format Structure for 1BASEMHS, baseHeader .............................. 765
261 Data Format Structure for 1BASEMHS, ScanTime ............................... 765
262 Data Format Structure for 1BASESAPHIR, SAPHIR base ......................... 773
263 Data Format Structure for 1BASESAPHIR, SAPHIR base ......................... 774
264 Data Format Structure for 1BASESAPHIR, baseHeader .......................... 774
265 Data Format Structure for 1BASESAPHIR, ScanTime ............................ 775
266 Data Format Structure for 1BASEATMS, ATMS base ............................ 784
267 Data Format Structure for 1BASEATMS, ATMS base ............................ 785
268 Data Format Structure for 1BASEATMS, ATMS base ............................ 786
269 Data Format Structure for 1BASEATMS, ScanTime .............................. 786
270 Data Format Structure for 1BGMI, GMI Brightness Temperatures ................. 796
271 Data Format Structure for 1BGMI, S1 ........................................... 797
272 Data Format Structure for 1BGMI, S2 ........................................... 798
273 Data Format Structure for 1BGMI, S1, ScanTime ................................ 798
274 Data Format Structure for 1BGMI, S1, scanStatus ................................ 799
275 Data Format Structure for 1BGMI, S1, sampleHeader ............................ 799
276 Data Format Structure for 1BGMI, S1, navigation ................................ 800
277 Data Format Structure for 1BGMI, S1, calibration ............................... 801
278 Data Format Structure for 1BGMI, S1, calCounts ................................ 801
279 Data Format Structure for 1BGMI, S1, sunData .................................. 802
280 Data Format Structure for 1BGMI, S2, ScanTime ................................ 802
281 Data Format Structure for 1BGMI, S2, scanStatus ................................ 803
282 Data Format Structure for 1BGMI, S2, sampleHeader ............................ 804
283 Data Format Structure for 1BGMI, S2, navigation ................................ 805
LIST OF FIGURES

284 Data Format Structure for 1BGMI, S2, calibration ................. 806
285 Data Format Structure for 1BGMI, S2, calCounts .................. 806
286 Data Format Structure for 1BGMI, S2, sunData .................... 807
287 Data Format Structure for 1BTMI, TMI unpacked packet data ....... 845
288 Data Format Structure for 1BTMI, S1 ............................ 845
289 Data Format Structure for 1BTMI, S2 ............................ 846
290 Data Format Structure for 1BTMI, S3 ............................ 847
291 Data Format Structure for 1BTMI, S1, ScanTime .................. 847
292 Data Format Structure for 1BTMI, S1, scanStatus ................. 848
293 Data Format Structure for 1BTMI, S1, navigation ................ 849
294 Data Format Structure for 1BTMI, S1, calibration ................. 849
295 Data Format Structure for 1BTMI, S1, calCounts .................. 850
296 Data Format Structure for 1BTMI, S1, sunData .................... 850
297 Data Format Structure for 1BTMI, S2, ScanTime .................. 850
298 Data Format Structure for 1BTMI, S2, scanStatus ................. 851
299 Data Format Structure for 1BTMI, S2, navigation ................ 852
300 Data Format Structure for 1BTMI, S2, calibration ................. 853
301 Data Format Structure for 1BTMI, S2, calCounts .................. 853
302 Data Format Structure for 1BTMI, S2, sunData .................... 854
303 Data Format Structure for 1BTMI, S3, ScanTime .................. 854
304 Data Format Structure for 1BTMI, S3, scanStatus ................. 855
305 Data Format Structure for 1BTMI, S3, navigation ................ 856
306 Data Format Structure for 1BTMI, S3, calibration ................. 856
307 Data Format Structure for 1BTMI, S3, calCounts .................. 857
308 Data Format Structure for 1BTMI, S3, sunData .................... 857
309 Data Format Structure for 1BVIRS, VIRS Radiance ............... 910
310 Data Format Structure for 1BVIRS, ScanTime .................... 911
311 Data Format Structure for 1BVIRS, scanStatus ................... 911
312 Data Format Structure for 1BVIRS, navigation ................... 912
313 Data Format Structure for 1BVIRS, solarCal ...................... 912
314 Data Format Structure for 1CTMI, GPM Common Calibrated Brightness Temperature 930
315 Data Format Structure for 1CTMI, S1 ............................ 930
316 Data Format Structure for 1CTMI, S2 ............................ 931
LIST OF FIGURES

317 Data Format Structure for 1CTMI, S3 ............................................ 931
318 Data Format Structure for 1CTMI, S1, ScanTime ................................. 932
319 Data Format Structure for 1CTMI, S1, SCstatus ................................. 932
320 Data Format Structure for 1CTMI, S2, ScanTime ................................. 932
321 Data Format Structure for 1CTMI, S2, SCstatus ................................. 933
322 Data Format Structure for 1CTMI, S3, ScanTime ................................. 933
323 Data Format Structure for 1CTMI, S3, SCstatus ................................. 933
324 Data Format Structure for 1CGMI, GPM Common Calibrated Brightness Temperature ................................................................. 950
325 Data Format Structure for 1CGMI, S1 ................................................ 951
326 Data Format Structure for 1CGMI, S2 ................................................ 951
327 Data Format Structure for 1CGMI, S1, ScanTime ................................ 952
328 Data Format Structure for 1CGMI, S1, SCstatus ................................ 952
329 Data Format Structure for 1CGMI, S2, ScanTime ................................ 952
330 Data Format Structure for 1CGMI, S2, SCstatus ................................ 953
331 Data Format Structure for 1CSSMI, Common Calibrated Brightness Temperature ................................................................. 966
332 Data Format Structure for 1CSSMI, S1 ................................................ 967
333 Data Format Structure for 1CSSMI, S2 ................................................ 967
334 Data Format Structure for 1CSSMI, S1, ScanTime ................................ 968
335 Data Format Structure for 1CSSMI, S1, SCstatus ................................ 968
336 Data Format Structure for 1CSSMI, S2, ScanTime ................................ 968
337 Data Format Structure for 1CSSMI, S2, SCstatus ................................ 969
338 Data Format Structure for 1CSSMIS, Common Calibrated Brightness Temperature ................................................................. 983
339 Data Format Structure for 1CSSMIS, S1 ............................................. 983
340 Data Format Structure for 1CSSMIS, S2 ............................................. 984
341 Data Format Structure for 1CSSMIS, S3 ............................................. 984
342 Data Format Structure for 1CSSMIS, S4 ............................................. 985
343 Data Format Structure for 1CSSMIS, S1, ScanTime ............................... 985
344 Data Format Structure for 1CSSMIS, S1, SCstatus ............................... 985
345 Data Format Structure for 1CSSMIS, S2, ScanTime ............................... 986
346 Data Format Structure for 1CSSMIS, S2, SCstatus ............................... 986
347 Data Format Structure for 1CSSMIS, S3, ScanTime ............................... 986
348 Data Format Structure for 1CSSMIS, S3, SCstatus ............................... 987
349 Data Format Structure for 1CSSMIS, S4, ScanTime ............................... 987
LIST OF FIGURES

350  Data Format Structure for 1CSSMIS, S4, SCstatus .......................... 987
351  Data Format Structure for 1ICAMSRE, Common Calibrated Brightness Temperature 1014
352  Data Format Structure for 1ICAMSRE, S1 ........................................ 1014
353  Data Format Structure for 1ICAMSRE, S2 ........................................ 1015
354  Data Format Structure for 1ICAMSRE, S3 ........................................ 1015
355  Data Format Structure for 1ICAMSRE, S4 ........................................ 1016
356  Data Format Structure for 1ICAMSRE, S5 ........................................ 1016
357  Data Format Structure for 1ICAMSRE, S6 ........................................ 1017
358  Data Format Structure for 1ICAMSRE, S1, ScanTime .......................... 1017
359  Data Format Structure for 1ICAMSRE, S1, SCstatus .......................... 1017
360  Data Format Structure for 1ICAMSRE, S2, ScanTime .......................... 1018
361  Data Format Structure for 1ICAMSRE, S2, SCstatus .......................... 1018
362  Data Format Structure for 1ICAMSRE, S3, ScanTime .......................... 1018
363  Data Format Structure for 1ICAMSRE, S3, SCstatus .......................... 1019
364  Data Format Structure for 1ICAMSRE, S4, ScanTime .......................... 1019
365  Data Format Structure for 1ICAMSRE, S4, SCstatus .......................... 1019
366  Data Format Structure for 1ICAMSRE, S5, ScanTime .......................... 1020
367  Data Format Structure for 1ICAMSRE, S5, SCstatus .......................... 1021
368  Data Format Structure for 1ICAMSRE, S6, ScanTime .......................... 1021
369  Data Format Structure for 1ICAMSRE, S6, SCstatus .......................... 1021
370  Data Format Structure for 1ICAMSRE2, Common Calibrated Brightness Temperature 1053
371  Data Format Structure for 1ICAMSRE2, S1 ........................................ 1053
372  Data Format Structure for 1ICAMSRE2, S2 ........................................ 1054
373  Data Format Structure for 1ICAMSRE2, S3 ........................................ 1054
374  Data Format Structure for 1ICAMSRE2, S4 ........................................ 1055
375  Data Format Structure for 1ICAMSRE2, S5 ........................................ 1055
376  Data Format Structure for 1ICAMSRE2, S6 ........................................ 1056
377  Data Format Structure for 1ICAMSRE2, S1, ScanTime .......................... 1056
378  Data Format Structure for 1ICAMSRE2, S1, SCstatus .......................... 1056
379  Data Format Structure for 1ICAMSRE2, S2, ScanTime .......................... 1057
380  Data Format Structure for 1ICAMSRE2, S2, SCstatus .......................... 1057
381  Data Format Structure for 1ICAMSRE2, S3, ScanTime .......................... 1057
382  Data Format Structure for 1ICAMSRE2, S3, SCstatus .......................... 1058
LIST OF FIGURES

383 Data Format Structure for ICAMSR2, S4, ScanTime ............... 1058
384 Data Format Structure for ICAMSR2, S4, SCstatus ............... 1058
385 Data Format Structure for ICAMSR2, S5, ScanTime ............... 1059
386 Data Format Structure for ICAMSR2, S5, SCstatus ............... 1060
387 Data Format Structure for ICAMSR2, S6, ScanTime ............... 1060
388 Data Format Structure for ICAMSR2, S6, SCstatus ............... 1060
389 Data Format Structure for ICWIND, Common Calibrated Brightness Temperature 1090
390 Data Format Structure for ICWIND, S1 .......................... 1090
391 Data Format Structure for ICWIND, ScanTime ................... 1091
392 Data Format Structure for ICWIND, SCstatus ................... 1091
393 Data Format Structure for ICMHS, Common Calibrated Brightness Temperature 1099
394 Data Format Structure for ICMHS, S1 .......................... 1100
395 Data Format Structure for ICMHS, ScanTime ................... 1100
396 Data Format Structure for ICMHS, SCstatus ................... 1100
397 Data Format Structure for ICSAPPHIR, Common Calibrated Brightness Temperature 1108
398 Data Format Structure for ICSAPPHIR, S1 .......................... 1108
399 Data Format Structure for ICSAPPHIR, ScanTime ................ 1108
400 Data Format Structure for ICSAPPHIR, SCstatus ................ 1109
401 Data Format Structure for ICATMS, Common Calibrated Brightness Temperature 1118
402 Data Format Structure for ICATMS, S1 .......................... 1118
403 Data Format Structure for ICATMS, S2 .......................... 1119
404 Data Format Structure for ICATMS, S3 .......................... 1119
405 Data Format Structure for ICATMS, S4 .......................... 1120
406 Data Format Structure for ICATMS, S1, ScanTime ............... 1120
407 Data Format Structure for ICATMS, S1, SCstatus ............... 1120
408 Data Format Structure for ICATMS, S2, ScanTime ............... 1121
409 Data Format Structure for ICATMS, S2, SCstatus ............... 1121
410 Data Format Structure for ICATMS, S3, ScanTime ............... 1121
411 Data Format Structure for ICATMS, S3, SCstatus ............... 1122
412 Data Format Structure for ICATMS, S4, ScanTime ............... 1122
413 Data Format Structure for ICATMS, S4, SCstatus ............... 1122
414 Data Format Structure for ICAMSUB, Common Calibrated Brightness Temperature 1143
415 Data Format Structure for ICAMSUB, S1 .......................... 1144
LIST OF FIGURES

416 Data Format Structure for 1CAMSUB, ScanTime .......................... 1144
417 Data Format Structure for 1CAMSUB, SCstatus .......................... 1144
418 Data Format Structure for 2AGPROFGMI, Radiometer Profiling ............. 1153
419 Data Format Structure for 2AGPROFGMI, Radiometer Profiling ............. 1154
420 Data Format Structure for 2AGPROFGMI, GprofDHeadr ..................... 1154
421 Data Format Structure for 2AGPROFGMI, ScanTime ........................ 1154
422 Data Format Structure for 2AGPROFGMI, SCstatus ........................ 1155
423 Data Format Structure for 2AGPROFTMI, Radiometer Profiling ............... 1166
424 Data Format Structure for 2AGPROFTMI, Radiometer Profiling ............... 1167
425 Data Format Structure for 2AGPROFTMI, GprofDHeadr ..................... 1167
426 Data Format Structure for 2AGPROFTMI, ScanTime ........................ 1167
427 Data Format Structure for 2AGPROFTMI, SCstatus ........................ 1168
428 Data Format Structure for 2AGPROFSSMI, Radiometer Profiling ............... 1179
429 Data Format Structure for 2AGPROFSSMI, Radiometer Profiling ............... 1180
430 Data Format Structure for 2AGPROFSSMI, GprofDHeadr ..................... 1180
431 Data Format Structure for 2AGPROFSSMI, ScanTime ........................ 1180
432 Data Format Structure for 2AGPROFSSMI, SCstatus ........................ 1181
433 Data Format Structure for 2AGPROFSSMIS, Radiometer Profiling .............. 1192
434 Data Format Structure for 2AGPROFSSMIS, Radiometer Profiling .............. 1193
435 Data Format Structure for 2AGPROFSSMIS, GprofDHeadr ..................... 1193
436 Data Format Structure for 2AGPROFSSMIS, ScanTime ........................ 1193
437 Data Format Structure for 2AGPROFSSMIS, SCstatus ........................ 1194
438 Data Format Structure for 2AGPROFAMSRE, Radiometer Profiling ............ 1205
439 Data Format Structure for 2AGPROFAMSRE, Radiometer Profiling ............ 1206
440 Data Format Structure for 2AGPROFAMSRE, GprofDHeadr ..................... 1206
441 Data Format Structure for 2AGPROFAMSRE, ScanTime ........................ 1206
442 Data Format Structure for 2AGPROFAMSRE, SCstatus ........................ 1207
443 Data Format Structure for 2AGPROFAMSRS2, Radiometer Profiling ............ 1219
444 Data Format Structure for 2AGPROFAMSRS2, Radiometer Profiling ............ 1220
445 Data Format Structure for 2AGPROFAMSRS2, GprofDHeadr ..................... 1220
446 Data Format Structure for 2AGPROFAMSRS2, ScanTime ........................ 1220
447 Data Format Structure for 2AGPROFAMSRS2, SCstatus ........................ 1221
448 Data Format Structure for 2AGPROFWIND, Radiometer Profiling .............. 1232
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>482</td>
<td>Data Format Structure for 1BKa, Ka Power</td>
<td>1329</td>
</tr>
<tr>
<td>483</td>
<td>Data Format Structure for 1BKa, MS</td>
<td>1330</td>
</tr>
<tr>
<td>484</td>
<td>Data Format Structure for 1BKa, HS</td>
<td>1330</td>
</tr>
<tr>
<td>485</td>
<td>Data Format Structure for 1BKa, MS, ScanTime</td>
<td>1331</td>
</tr>
<tr>
<td>486</td>
<td>Data Format Structure for 1BKa, MS, scanStatus</td>
<td>1331</td>
</tr>
<tr>
<td>487</td>
<td>Data Format Structure for 1BKa, MS, navigation</td>
<td>1332</td>
</tr>
<tr>
<td>488</td>
<td>Data Format Structure for 1BKa, MS, rayPointing</td>
<td>1332</td>
</tr>
<tr>
<td>489</td>
<td>Data Format Structure for 1BKa, MS, HouseKeeping</td>
<td>1333</td>
</tr>
<tr>
<td>490</td>
<td>Data Format Structure for 1BKa, MS, VertLocate</td>
<td>1334</td>
</tr>
<tr>
<td>491</td>
<td>Data Format Structure for 1BKa, MS, Calibration</td>
<td>1334</td>
</tr>
<tr>
<td>492</td>
<td>Data Format Structure for 1BKa, MS, Transmitter</td>
<td>1335</td>
</tr>
<tr>
<td>493</td>
<td>Data Format Structure for 1BKa, MS, Receiver</td>
<td>1335</td>
</tr>
<tr>
<td>494</td>
<td>Data Format Structure for 1BKa, HS, ScanTime</td>
<td>1336</td>
</tr>
<tr>
<td>495</td>
<td>Data Format Structure for 1BKa, HS, scanStatus</td>
<td>1337</td>
</tr>
<tr>
<td>496</td>
<td>Data Format Structure for 1BKa, HS, navigation</td>
<td>1337</td>
</tr>
<tr>
<td>497</td>
<td>Data Format Structure for 1BKa, HS, rayPointing</td>
<td>1338</td>
</tr>
<tr>
<td>498</td>
<td>Data Format Structure for 1BKa, HS, HouseKeeping</td>
<td>1338</td>
</tr>
<tr>
<td>499</td>
<td>Data Format Structure for 1BKa, HS, VertLocate</td>
<td>1339</td>
</tr>
<tr>
<td>500</td>
<td>Data Format Structure for 1BKa, HS, Calibration</td>
<td>1339</td>
</tr>
<tr>
<td>501</td>
<td>Data Format Structure for 1BKa, HS, Transmitter</td>
<td>1340</td>
</tr>
<tr>
<td>502</td>
<td>Data Format Structure for 1BKa, HS, Receiver</td>
<td>1340</td>
</tr>
<tr>
<td>503</td>
<td>Data Format Structure for 1BPR, PR Power</td>
<td>1391</td>
</tr>
<tr>
<td>504</td>
<td>Data Format Structure for 1BPR, ScanTime</td>
<td>1391</td>
</tr>
<tr>
<td>505</td>
<td>Data Format Structure for 1BPR, scanStatus</td>
<td>1392</td>
</tr>
<tr>
<td>506</td>
<td>Data Format Structure for 1BPR, navigation</td>
<td>1392</td>
</tr>
<tr>
<td>507</td>
<td>Data Format Structure for 1BPR, rayPointing</td>
<td>1393</td>
</tr>
<tr>
<td>508</td>
<td>Data Format Structure for 1BPR, HouseKeeping</td>
<td>1393</td>
</tr>
<tr>
<td>509</td>
<td>Data Format Structure for 1BPR, VertLocate</td>
<td>1394</td>
</tr>
<tr>
<td>510</td>
<td>Data Format Structure for 1BPR, Calibration</td>
<td>1394</td>
</tr>
<tr>
<td>511</td>
<td>Data Format Structure for 1BPR, Transmitter</td>
<td>1395</td>
</tr>
<tr>
<td>512</td>
<td>Data Format Structure for 1BPR, Receiver</td>
<td>1395</td>
</tr>
<tr>
<td>513</td>
<td>Data Format Structure for 2AKu, Ku precipitation</td>
<td>1422</td>
</tr>
<tr>
<td>514</td>
<td>Data Format Structure for 2AKu, ScanTime</td>
<td>1423</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>515</td>
<td>Data Format Structure for 2AKu, scanStatus</td>
<td>1423</td>
</tr>
<tr>
<td>516</td>
<td>Data Format Structure for 2AKu, navigation</td>
<td>1424</td>
</tr>
<tr>
<td>517</td>
<td>Data Format Structure for 2AKu, PRE</td>
<td>1425</td>
</tr>
<tr>
<td>518</td>
<td>Data Format Structure for 2AKu, VER</td>
<td>1425</td>
</tr>
<tr>
<td>519</td>
<td>Data Format Structure for 2AKu, CSF</td>
<td>1426</td>
</tr>
<tr>
<td>520</td>
<td>Data Format Structure for 2AKu, SRT</td>
<td>1426</td>
</tr>
<tr>
<td>521</td>
<td>Data Format Structure for 2AKu, DSD</td>
<td>1427</td>
</tr>
<tr>
<td>522</td>
<td>Data Format Structure for 2AKu, Experimental</td>
<td>1427</td>
</tr>
<tr>
<td>523</td>
<td>Data Format Structure for 2AKa, Ka precipitation</td>
<td>1467</td>
</tr>
<tr>
<td>524</td>
<td>Data Format Structure for 2AKa, FLG</td>
<td>1429</td>
</tr>
<tr>
<td>525</td>
<td>Data Format Structure for 2AKa, MS</td>
<td>1467</td>
</tr>
<tr>
<td>526</td>
<td>Data Format Structure for 2AKa, MS, ScanTime</td>
<td>1467</td>
</tr>
<tr>
<td>527</td>
<td>Data Format Structure for 2AKa, MS, PRE</td>
<td>1469</td>
</tr>
<tr>
<td>528</td>
<td>Data Format Structure for 2AKa, MS, VER</td>
<td>1469</td>
</tr>
<tr>
<td>529</td>
<td>Data Format Structure for 2AKa, MS, CSF</td>
<td>1471</td>
</tr>
<tr>
<td>530</td>
<td>Data Format Structure for 2AKa, MS, SRT</td>
<td>1471</td>
</tr>
<tr>
<td>531</td>
<td>Data Format Structure for 2AKa, MS, DSD</td>
<td>1471</td>
</tr>
<tr>
<td>532</td>
<td>Data Format Structure for 2AKa, MS, Experimental</td>
<td>1472</td>
</tr>
<tr>
<td>533</td>
<td>Data Format Structure for 2AKa, MS, FLG</td>
<td>1473</td>
</tr>
<tr>
<td>534</td>
<td>Data Format Structure for 2AKa, HS, ScanTime</td>
<td>1474</td>
</tr>
<tr>
<td>535</td>
<td>Data Format Structure for 2AKa, HS, scanStatus</td>
<td>1474</td>
</tr>
<tr>
<td>536</td>
<td>Data Format Structure for 2AKa, HS, navigation</td>
<td>1475</td>
</tr>
<tr>
<td>537</td>
<td>Data Format Structure for 2AKa, HS, PRE</td>
<td>1476</td>
</tr>
<tr>
<td>538</td>
<td>Data Format Structure for 2AKa, HS, VER</td>
<td>1476</td>
</tr>
<tr>
<td>539</td>
<td>Data Format Structure for 2AKa, HS, CSF</td>
<td>1477</td>
</tr>
<tr>
<td>540</td>
<td>Data Format Structure for 2AKa, HS, SRT</td>
<td>1477</td>
</tr>
<tr>
<td>541</td>
<td>Data Format Structure for 2AKa, HS, DSD</td>
<td>1477</td>
</tr>
<tr>
<td>542</td>
<td>Data Format Structure for 2AKa, HS, Experimental</td>
<td>1478</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

548 Data Format Structure for 2AKa, HS, SLV ..................... 1479
549 Data Format Structure for 2AKa, HS, FLG ..................... 1479
550 Data Format Structure for 2ADPR, DPR precipitation .......... 1553
551 Data Format Structure for 2ADPR, NS ........................ 1553
552 Data Format Structure for 2ADPR, MS ........................ 1554
553 Data Format Structure for 2ADPR, HS ........................ 1555
554 Data Format Structure for 2ADPR, NS, ScanTime .............. 1555
555 Data Format Structure for 2ADPR, NS, scanStatus .......... 1556
556 Data Format Structure for 2ADPR, NS, navigation .......... 1556
557 Data Format Structure for 2ADPR, NS, PRE ................... 1557
558 Data Format Structure for 2ADPR, NS, VER ................... 1557
559 Data Format Structure for 2ADPR, NS, CSF ................... 1558
560 Data Format Structure for 2ADPR, NS, SRT ................... 1558
561 Data Format Structure for 2ADPR, NS, DSD ................. 1559
562 Data Format Structure for 2ADPR, NS, Experimental ........ 1559
563 Data Format Structure for 2ADPR, NS, SLV ................. 1560
564 Data Format Structure for 2ADPR, NS, FLG ................... 1560
565 Data Format Structure for 2ADPR, MS, ScanTime .......... 1561
566 Data Format Structure for 2ADPR, MS, scanStatus .......... 1561
567 Data Format Structure for 2ADPR, MS, navigation .......... 1562
568 Data Format Structure for 2ADPR, MS, PRE .................. 1563
569 Data Format Structure for 2ADPR, MS, VER .................. 1563
570 Data Format Structure for 2ADPR, MS, CSF .................. 1564
571 Data Format Structure for 2ADPR, MS, SRT .................. 1564
572 Data Format Structure for 2ADPR, MS, DSD .................. 1565
573 Data Format Structure for 2ADPR, MS, Experimental ....... 1566
574 Data Format Structure for 2ADPR, MS, SLV .................. 1566
575 Data Format Structure for 2ADPR, MS, FLG .................. 1566
576 Data Format Structure for 2ADPR, TRG ..................... 1567
577 Data Format Structure for 2ADPR, MS, TRG .................. 1568
578 Data Format Structure for 2ADPR, HS, ScanTime .......... 1568
579 Data Format Structure for 2ADPR, HS, scanStatus .......... 1568
580 Data Format Structure for 2ADPR, HS, navigation .......... 1569
LIST OF FIGURES

581 Data Format Structure for 2ADPR, HS, PRE .................................. 1570
582 Data Format Structure for 2ADPR, HS, VER .................................. 1570
583 Data Format Structure for 2ADPR, HS, CSF .................................. 1571
584 Data Format Structure for 2ADPR, HS, SRT .................................. 1572
585 Data Format Structure for 2ADPR, HS, DSD .................................. 1572
586 Data Format Structure for 2ADPR, HS, Experimental .......................... 1573
587 Data Format Structure for 2ADPR, HS, SLV .................................. 1573
588 Data Format Structure for 2ADPR, HS, FLG .................................. 1573
589 Data Format Structure for 2APR, PR precipitation ................................ 1687
590 Data Format Structure for 2APR, ScanTime .................................. 1688
591 Data Format Structure for 2APR, scanStatus .................................. 1688
592 Data Format Structure for 2APR, navigation .................................. 1689
593 Data Format Structure for 2APR, PRE ........................................ 1690
594 Data Format Structure for 2APR, VER ........................................ 1690
595 Data Format Structure for 2APR, CSF ........................................ 1691
596 Data Format Structure for 2APR, SRT ........................................ 1691
597 Data Format Structure for 2APR, DSD ........................................ 1692
598 Data Format Structure for 2APR, Experimental ................................ 1692
599 Data Format Structure for 2APR, SLV ........................................ 1693
600 Data Format Structure for 2APR, FLG ........................................ 1694
601 Data Format Structure for 3DPR, DPR Full Product ............................ 1734
602 Data Format Structure for 3DPR, G1, G1 ..................................... 1735
603 Data Format Structure for 3DPR, G1, G1 ..................................... 1736
604 Data Format Structure for 3DPR, G1 ......................................... 1737
605 Data Format Structure for 3DPR, G2, G2 ..................................... 1737
606 Data Format Structure for 3DPR, G2 ......................................... 1738
607 Data Format Structure for 3DPR, G1, precipRate ................................ 1738
608 Data Format Structure for 3DPR, G1, rainRate ................................ 1739
609 Data Format Structure for 3DPR, G1, snowRate ................................ 1739
610 Data Format Structure for 3DPR, G1, flagHeavyIcePrecip ..................... 1739
611 Data Format Structure for 3DPR, G1, mixedPhRate ................................ 1739
612 Data Format Structure for 3DPR, G1, precipRateESurface ...................... 1739
613 Data Format Structure for 3DPR, G1, precipRateESurface2 .................... 1740
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>614</td>
<td>Data Format Structure for 3DPR, G1, precipRateNearSurface</td>
<td>1740</td>
</tr>
<tr>
<td>615</td>
<td>Data Format Structure for 3DPR, G1, rainRateNearSurface</td>
<td>1740</td>
</tr>
<tr>
<td>616</td>
<td>Data Format Structure for 3DPR, G1, snowRateNearSurface</td>
<td>1741</td>
</tr>
<tr>
<td>617</td>
<td>Data Format Structure for 3DPR, G1, mixedPhRateNearSurface</td>
<td>1741</td>
</tr>
<tr>
<td>618</td>
<td>Data Format Structure for 3DPR, G1, precipWaterIntegrated</td>
<td>1741</td>
</tr>
<tr>
<td>619</td>
<td>Data Format Structure for 3DPR, G1, precipIceIntegrated</td>
<td>1741</td>
</tr>
<tr>
<td>620</td>
<td>Data Format Structure for 3DPR, G1, precipRateAve24</td>
<td>1741</td>
</tr>
<tr>
<td>621</td>
<td>Data Format Structure for 3DPR, G1, zFactorCorrected</td>
<td>1742</td>
</tr>
<tr>
<td>622</td>
<td>Data Format Structure for 3DPR, G1, zFactorCorrectedESurface</td>
<td>1742</td>
</tr>
<tr>
<td>623</td>
<td>Data Format Structure for 3DPR, G1, zFactorCorrectedNearSurface</td>
<td>1742</td>
</tr>
<tr>
<td>624</td>
<td>Data Format Structure for 3DPR, G1, zFactorMeasuredNearSurface</td>
<td>1742</td>
</tr>
<tr>
<td>625</td>
<td>Data Format Structure for 3DPR, G1, zFactorCorrectedDPR</td>
<td>1742</td>
</tr>
<tr>
<td>626</td>
<td>Data Format Structure for 3DPR, G1, zFactorCorrectedESurfaceDPR</td>
<td>1743</td>
</tr>
<tr>
<td>627</td>
<td>Data Format Structure for 3DPR, G1, zFactorCorrectedNearSurfaceDPR</td>
<td>1743</td>
</tr>
<tr>
<td>628</td>
<td>Data Format Structure for 3DPR, G1, zFactorMeasured</td>
<td>1743</td>
</tr>
<tr>
<td>629</td>
<td>Data Format Structure for 3DPR, G1, dm</td>
<td>1743</td>
</tr>
<tr>
<td>630</td>
<td>Data Format Structure for 3DPR, G1, dBNw</td>
<td>1743</td>
</tr>
<tr>
<td>631</td>
<td>Data Format Structure for 3DPR, G1, epsilonDPR</td>
<td>1744</td>
</tr>
<tr>
<td>632</td>
<td>Data Format Structure for 3DPR, G1, epsilon</td>
<td>1744</td>
</tr>
<tr>
<td>633</td>
<td>Data Format Structure for 3DPR, G1, zeta</td>
<td>1744</td>
</tr>
<tr>
<td>634</td>
<td>Data Format Structure for 3DPR, G1, piaHB</td>
<td>1744</td>
</tr>
<tr>
<td>635</td>
<td>Data Format Structure for 3DPR, G1, piaHybrid</td>
<td>1744</td>
</tr>
<tr>
<td>636</td>
<td>Data Format Structure for 3DPR, G1, piaHybridDPR</td>
<td>1745</td>
</tr>
<tr>
<td>637</td>
<td>Data Format Structure for 3DPR, G1, piaSRT</td>
<td>1745</td>
</tr>
<tr>
<td>638</td>
<td>Data Format Structure for 3DPR, G1, piaSRTdpr</td>
<td>1745</td>
</tr>
<tr>
<td>639</td>
<td>Data Format Structure for 3DPR, G1, piaFinal</td>
<td>1745</td>
</tr>
<tr>
<td>640</td>
<td>Data Format Structure for 3DPR, G1, piaFinalDPR</td>
<td>1745</td>
</tr>
<tr>
<td>641</td>
<td>Data Format Structure for 3DPR, G1, piaFinalSubset</td>
<td>1746</td>
</tr>
<tr>
<td>642</td>
<td>Data Format Structure for 3DPR, G1, piaFinalDPRsubset</td>
<td>1746</td>
</tr>
<tr>
<td>643</td>
<td>Data Format Structure for 3DPR, G1, heightBB</td>
<td>1746</td>
</tr>
<tr>
<td>644</td>
<td>Data Format Structure for 3DPR, G1, heightBBnadir</td>
<td>1746</td>
</tr>
<tr>
<td>645</td>
<td>Data Format Structure for 3DPR, G1, BBwidthNadir</td>
<td>1746</td>
</tr>
<tr>
<td>646</td>
<td>Data Format Structure for 3DPR, G1, heightStormTop</td>
<td>1747</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>647</td>
<td>Data Format Structure for 3DPR, G1, BBwidth</td>
<td>1747</td>
</tr>
<tr>
<td>648</td>
<td>Data Format Structure for 3DPR, G1, observationCounts</td>
<td>1747</td>
</tr>
<tr>
<td>649</td>
<td>Data Format Structure for 3DPR, G1, precipRateLocalTime</td>
<td>1747</td>
</tr>
<tr>
<td>650</td>
<td>Data Format Structure for 3DPR, G1, DFRmNearSurface</td>
<td>1747</td>
</tr>
<tr>
<td>651</td>
<td>Data Format Structure for 3DPR, G1, DFRNearSurface</td>
<td>1747</td>
</tr>
<tr>
<td>652</td>
<td>Data Format Structure for 3DPR, G2, precipRate</td>
<td>1748</td>
</tr>
<tr>
<td>653</td>
<td>Data Format Structure for 3DPR, G2, rainRate</td>
<td>1748</td>
</tr>
<tr>
<td>654</td>
<td>Data Format Structure for 3DPR, G2, snowRate</td>
<td>1748</td>
</tr>
<tr>
<td>655</td>
<td>Data Format Structure for 3DPR, G2, flagHeavyIcePrecip</td>
<td>1748</td>
</tr>
<tr>
<td>656</td>
<td>Data Format Structure for 3DPR, G2, mixedPhRate</td>
<td>1749</td>
</tr>
<tr>
<td>657</td>
<td>Data Format Structure for 3DPR, G2, precipRateESurface</td>
<td>1749</td>
</tr>
<tr>
<td>658</td>
<td>Data Format Structure for 3DPR, G2, precipRateESurface2</td>
<td>1749</td>
</tr>
<tr>
<td>659</td>
<td>Data Format Structure for 3DPR, G2, precipRateNearSurface</td>
<td>1749</td>
</tr>
<tr>
<td>660</td>
<td>Data Format Structure for 3DPR, G2, rainRateNearSurface</td>
<td>1749</td>
</tr>
<tr>
<td>661</td>
<td>Data Format Structure for 3DPR, G2, snowRateNearSurface</td>
<td>1749</td>
</tr>
<tr>
<td>662</td>
<td>Data Format Structure for 3DPR, G2, mixedPhRateNearSurface</td>
<td>1749</td>
</tr>
<tr>
<td>663</td>
<td>Data Format Structure for 3DPR, G2, precipWaterIntegrated</td>
<td>1750</td>
</tr>
<tr>
<td>664</td>
<td>Data Format Structure for 3DPR, G2, precipIceIntegrated</td>
<td>1750</td>
</tr>
<tr>
<td>665</td>
<td>Data Format Structure for 3DPR, G2, precipRateAve24</td>
<td>1750</td>
</tr>
<tr>
<td>666</td>
<td>Data Format Structure for 3DPR, G2, zFactorCorrected</td>
<td>1751</td>
</tr>
<tr>
<td>667</td>
<td>Data Format Structure for 3DPR, G2, zFactorCorrectedESurface</td>
<td>1751</td>
</tr>
<tr>
<td>668</td>
<td>Data Format Structure for 3DPR, G2, zFactorCorrectedNearSurface</td>
<td>1751</td>
</tr>
<tr>
<td>669</td>
<td>Data Format Structure for 3DPR, G2, zFactorMeasuredNearSurface</td>
<td>1751</td>
</tr>
<tr>
<td>670</td>
<td>Data Format Structure for 3DPR, G2, zFactorCorrectedDPR</td>
<td>1751</td>
</tr>
<tr>
<td>671</td>
<td>Data Format Structure for 3DPR, G2, zFactorCorrectedESurfaceDPR</td>
<td>1751</td>
</tr>
<tr>
<td>672</td>
<td>Data Format Structure for 3DPR, G2, zFactorCorrectedNearSurfaceDPR</td>
<td>1751</td>
</tr>
<tr>
<td>673</td>
<td>Data Format Structure for 3DPR, G2, zFactorMeasured</td>
<td>1752</td>
</tr>
<tr>
<td>674</td>
<td>Data Format Structure for 3DPR, G2, dm</td>
<td>1752</td>
</tr>
<tr>
<td>675</td>
<td>Data Format Structure for 3DPR, G2, dBNw</td>
<td>1752</td>
</tr>
<tr>
<td>676</td>
<td>Data Format Structure for 3DPR, G2, epsilonDPR</td>
<td>1753</td>
</tr>
<tr>
<td>677</td>
<td>Data Format Structure for 3DPR, G2, epsilon</td>
<td>1753</td>
</tr>
<tr>
<td>678</td>
<td>Data Format Structure for 3DPR, G2, zeta</td>
<td>1753</td>
</tr>
<tr>
<td>679</td>
<td>Data Format Structure for 3DPR, G2, piaHB</td>
<td>1753</td>
</tr>
</tbody>
</table>
680 Data Format Structure for 3DPR, G2, piaHybrid ................. 1753
681 Data Format Structure for 3DPR, G2, piaHybridDPR .......... 1753
682 Data Format Structure for 3DPR, G2, piaSRT .................. 1753
683 Data Format Structure for 3DPR, G2, piaSRTdpr .......... 1754
684 Data Format Structure for 3DPR, G2, piaFinal ................ 1754
685 Data Format Structure for 3DPR, G2, piaFinalDPR .......... 1754
686 Data Format Structure for 3DPR, G2, heightBB ................ 1755
687 Data Format Structure for 3DPR, G2, heightStormTop ....... 1755
688 Data Format Structure for 3DPR, G2, BBwidth ................ 1755
689 Data Format Structure for 3DPR, G2, observationCounts .... 1756
690 Data Format Structure for 3DPR, G2, DFRmNearSurface .... 1756
691 Data Format Structure for 3DPR, G2, DFRNearSurface ..... 1756
692 Data Format Structure for 3DPRD, DPR Daily Product ....... 1831
693 Data Format Structure for 3DPRD, DPR Daily Product ....... 1832
694 Data Format Structure for 3DPRD, GridTimeAsc .............. 1832
695 Data Format Structure for 3DPRD, GridTimeDes .............. 1833
696 Data Format Structure for 3PR, PR Full Product .......... 1844
697 Data Format Structure for 3PR, G1, G1 ....................... 1845
698 Data Format Structure for 3PR, G1, G1 ....................... 1846
699 Data Format Structure for 3PR, G1 ....................... 1847
700 Data Format Structure for 3PR, G2, G2 ....................... 1847
701 Data Format Structure for 3PR, G2 ....................... 1848
702 Data Format Structure for 3PR, G1, precipRate ............. 1848
703 Data Format Structure for 3PR, G1, rainRate ............... 1849
704 Data Format Structure for 3PR, G1, snowRate ............... 1849
705 Data Format Structure for 3PR, G1, flagHeavyIcePrecip ...... 1849
706 Data Format Structure for 3PR, G1, mixedPhRate .......... 1849
707 Data Format Structure for 3PR, G1, precipRateESurface .... 1849
708 Data Format Structure for 3PR, G1, precipRateESurface2 .... 1850
709 Data Format Structure for 3PR, G1, precipRateNearSurface .. 1850
710 Data Format Structure for 3PR, G1, rainRateNearSurface ... 1850
711 Data Format Structure for 3PR, G1, snowRateNearSurface ... 1851
712 Data Format Structure for 3PR, G1, mixedPhRateNearSurface .. 1851
<table>
<thead>
<tr>
<th>Data Format Structure for 3PR, G1, precipWaterIntegrated</th>
<th>1851</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Format Structure for 3PR, G1, precipIceIntegrated</td>
<td>1851</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, precipRateAve24</td>
<td>1851</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, zFactorCorrected</td>
<td>1852</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, zFactorCorrectedESurface</td>
<td>1852</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, zFactorCorrectedNearSurface</td>
<td>1852</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, zFactorMeasuredNearSurface</td>
<td>1852</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, zFactorCorrectedDPR</td>
<td>1852</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, zFactorCorrectedESurfaceDPR</td>
<td>1853</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, zFactorCorrectedNearSurfaceDPR</td>
<td>1853</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, zFactorMeasured</td>
<td>1853</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, dm</td>
<td>1853</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, dBnW</td>
<td>1853</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, epsilonDPR</td>
<td>1854</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, epsilon</td>
<td>1854</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, zeta</td>
<td>1854</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, piaHB</td>
<td>1854</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, piaHybrid</td>
<td>1854</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, piaHybridDPR</td>
<td>1855</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, piaSRT</td>
<td>1855</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, piaSRDpr</td>
<td>1855</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, piaFinal</td>
<td>1855</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, piaFinalDPR</td>
<td>1855</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, piaFinalSubset</td>
<td>1856</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, piaFinalDPRsubset</td>
<td>1856</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, heightBB</td>
<td>1856</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, heightBBnadir</td>
<td>1856</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, BBwidthNadir</td>
<td>1856</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, heightStormTop</td>
<td>1857</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, BBwidth</td>
<td>1857</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, observationCounts</td>
<td>1857</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, precipRateLocalTime</td>
<td>1857</td>
</tr>
<tr>
<td>Data Format Structure for 3PR, G1, DFRmNearSurface</td>
<td>1857</td>
</tr>
</tbody>
</table>
746 Data Format Structure for 3PR, G1, DFRNearSurface ............... 1857
747 Data Format Structure for 3PR, G2, precipRate ..................... 1858
748 Data Format Structure for 3PR, G2, rainRate ........................ 1858
749 Data Format Structure for 3PR, G2, snowRate ......................... 1858
750 Data Format Structure for 3PR, G2, flagHeavyIcePrecip .............. 1858
751 Data Format Structure for 3PR, G2, mixedPhRate ...................... 1859
752 Data Format Structure for 3PR, G2, precipRateESurface ............... 1859
753 Data Format Structure for 3PR, G2, precipRateESurface2 ............. 1859
754 Data Format Structure for 3PR, G2, precipRateNearSurface .......... 1859
755 Data Format Structure for 3PR, G2, rainRateNearSurface ............. 1859
756 Data Format Structure for 3PR, G2, snowRateNearSurface ........... 1859
757 Data Format Structure for 3PR, G2, mixedPhRateNearSurface ......... 1859
758 Data Format Structure for 3PR, G2, precipWaterIntegrated .......... 1860
759 Data Format Structure for 3PR, G2, precipIceIntegrated .............. 1860
760 Data Format Structure for 3PR, G2, precipRateAve24 ................ 1860
761 Data Format Structure for 3PR, G2, zFactorCorrected ................ 1861
762 Data Format Structure for 3PR, G2, zFactorCorrectedESurface ....... 1861
763 Data Format Structure for 3PR, G2, zFactorCorrectedNearSurface ...... 1861
764 Data Format Structure for 3PR, G2, zFactorMeasuredNearSurface ...... 1861
765 Data Format Structure for 3PR, G2, zFactorCorrectedDPR ............. 1861
766 Data Format Structure for 3PR, G2, zFactorCorrectedESurfaceDPR .... 1861
767 Data Format Structure for 3PR, G2, zFactorCorrectedNearSurfaceDPR .. 1861
768 Data Format Structure for 3PR, G2, zFactorMeasured ................ 1862
769 Data Format Structure for 3PR, G2, dm .............................. 1862
770 Data Format Structure for 3PR, G2, dBnW ........................... 1862
771 Data Format Structure for 3PR, G2, epsilonDPR ........................ 1863
772 Data Format Structure for 3PR, G2, epsilon .......................... 1863
773 Data Format Structure for 3PR, G2, zeta .............................. 1863
774 Data Format Structure for 3PR, G2, piaHB ............................ 1863
775 Data Format Structure for 3PR, G2, piaHybrid ......................... 1863
776 Data Format Structure for 3PR, G2, piaHybridDPR ...................... 1863
777 Data Format Structure for 3PR, G2, piaSRT ............................ 1863
778 Data Format Structure for 3PR, G2, piaSRTdpr ........................ 1864
LIST OF FIGURES

779  Data Format Structure for 3PR, G2, piaFinal ........................................ 1864
780  Data Format Structure for 3PR, G2, piaFinalDPR ...................................... 1864
781  Data Format Structure for 3PR, G2, heightBB ........................................... 1865
782  Data Format Structure for 3PR, G2, heightStormTop .................................... 1865
783  Data Format Structure for 3PR, G2, BBwidth ............................................. 1865
784  Data Format Structure for 3PR, G2, observationCounts .................................. 1866
785  Data Format Structure for 3PR, G2, DFRmNearSurface .................................. 1866
786  Data Format Structure for 3PR, G2, DFRNearSurface .................................... 1866
787  Data Format Structure for 3PRD, PR Daily Product ..................................... 1941
788  Data Format Structure for 3PRD, PR Daily Product ..................................... 1942
789  Data Format Structure for 3PRD, GridTimeAsc .......................................... 1942
790  Data Format Structure for 3PRD, GridTimeDes .......................................... 1943
791  Data Format Structure for 2BCMB, Level-2 DPR and GMI Combined ..................... 1952
792  Data Format Structure for 2BCMB, NS, NS .................................................... 1953
793  Data Format Structure for 2BCMB, NS ......................................................... 1954
794  Data Format Structure for 2BCMB, MS, MS ................................................... 1955
795  Data Format Structure for 2BCMB, MS ......................................................... 1956
796  Data Format Structure for 2BCMB, NS, ScanTime ....................................... 1957
797  Data Format Structure for 2BCMB, NS, scanStatus ...................................... 1957
798  Data Format Structure for 2BCMB, NS, navigation ..................................... 1958
799  Data Format Structure for 2BCMB, NS, Input ............................................ 1959
800  Data Format Structure for 2BCMB, NS, aPriori ......................................... 1959
801  Data Format Structure for 2BCMB, NS, FLG ............................................... 1959
802  Data Format Structure for 2BCMB, MS, ScanTime ...................................... 1960
803  Data Format Structure for 2BCMB, MS, scanStatus .................................... 1960
804  Data Format Structure for 2BCMB, MS, navigation .................................... 1961
805  Data Format Structure for 2BCMB, MS, Input ............................................ 1962
806  Data Format Structure for 2BCMB, MS, aPriori ........................................... 1963
807  Data Format Structure for 2BCMB, MS, FLG ............................................... 1963
808  Data Format Structure for 3CMB, Combined precipitation ................................ 2006
809  Data Format Structure for 3CMB, G1 ......................................................... 2007
810  Data Format Structure for 3CMB, G2 ......................................................... 2008
811  Data Format Structure for 3CMB, G1, precipTotRate .................................... 2008
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>812</td>
<td>Data Format Structure for 3CMB, G1, precipLiqRate</td>
<td>2008</td>
</tr>
<tr>
<td>813</td>
<td>Data Format Structure for 3CMB, G1, precipTotWaterContent</td>
<td>2009</td>
</tr>
<tr>
<td>814</td>
<td>Data Format Structure for 3CMB, G1, precipLiqWaterContent</td>
<td>2009</td>
</tr>
<tr>
<td>815</td>
<td>Data Format Structure for 3CMB, G1, cloudLiqWaterContent</td>
<td>2009</td>
</tr>
<tr>
<td>816</td>
<td>Data Format Structure for 3CMB, G1, precipTotDm</td>
<td>2009</td>
</tr>
<tr>
<td>817</td>
<td>Data Format Structure for 3CMB, G1, precipTotLogNw</td>
<td>2009</td>
</tr>
<tr>
<td>818</td>
<td>Data Format Structure for 3CMB, G1, surfPrecipTotRateDiurnal</td>
<td>2009</td>
</tr>
<tr>
<td>819</td>
<td>Data Format Structure for 3CMB, G2, precipTotRate</td>
<td>2010</td>
</tr>
<tr>
<td>820</td>
<td>Data Format Structure for 3CMB, G2, precipLiqRate</td>
<td>2011</td>
</tr>
<tr>
<td>821</td>
<td>Data Format Structure for 3CMB, G2, precipTotWaterContent</td>
<td>2011</td>
</tr>
<tr>
<td>822</td>
<td>Data Format Structure for 3CMB, G2, precipLiqWaterContent</td>
<td>2011</td>
</tr>
<tr>
<td>823</td>
<td>Data Format Structure for 3CMB, G2, cloudLiqWaterContent</td>
<td>2012</td>
</tr>
<tr>
<td>824</td>
<td>Data Format Structure for 3CMB, G2, precipTotDm</td>
<td>2012</td>
</tr>
<tr>
<td>825</td>
<td>Data Format Structure for 3CMB, G2, precipTotLogNw</td>
<td>2012</td>
</tr>
<tr>
<td>826</td>
<td>Data Format Structure for 3CMB, G2, surfPrecipTotRateDiurnal</td>
<td>2012</td>
</tr>
<tr>
<td>827</td>
<td>Data Format Structure for 2BCMBT, Level-2 PR and TMI Combined</td>
<td>2031</td>
</tr>
<tr>
<td>828</td>
<td>Data Format Structure for 2BCMBT, NS</td>
<td>2032</td>
</tr>
<tr>
<td>829</td>
<td>Data Format Structure for 2BCMBT, NS</td>
<td>2033</td>
</tr>
<tr>
<td>830</td>
<td>Data Format Structure for 2BCMBT, ScanTime</td>
<td>2034</td>
</tr>
<tr>
<td>831</td>
<td>Data Format Structure for 2BCMBT, scanStatus</td>
<td>2034</td>
</tr>
<tr>
<td>832</td>
<td>Data Format Structure for 2BCMBT, navigation</td>
<td>2035</td>
</tr>
<tr>
<td>833</td>
<td>Data Format Structure for 2BCMBT, Input</td>
<td>2036</td>
</tr>
<tr>
<td>834</td>
<td>Data Format Structure for 2BCMBT, aPriori</td>
<td>2036</td>
</tr>
<tr>
<td>835</td>
<td>Data Format Structure for 2BCMBT, FLG</td>
<td>2036</td>
</tr>
<tr>
<td>836</td>
<td>Data Format Structure for 3CMBT, Combined precipitation</td>
<td>2059</td>
</tr>
<tr>
<td>837</td>
<td>Data Format Structure for 3CMBT, G1</td>
<td>2060</td>
</tr>
<tr>
<td>838</td>
<td>Data Format Structure for 3CMBT, G2</td>
<td>2061</td>
</tr>
<tr>
<td>839</td>
<td>Data Format Structure for 3CMBT, G1, precipTotRate</td>
<td>2061</td>
</tr>
<tr>
<td>840</td>
<td>Data Format Structure for 3CMBT, G1, precipLiqRate</td>
<td>2061</td>
</tr>
<tr>
<td>841</td>
<td>Data Format Structure for 3CMBT, G1, precipTotWaterContent</td>
<td>2062</td>
</tr>
<tr>
<td>842</td>
<td>Data Format Structure for 3CMBT, G1, precipLiqWaterContent</td>
<td>2062</td>
</tr>
<tr>
<td>843</td>
<td>Data Format Structure for 3CMBT, G1, cloudLiqWaterContent</td>
<td>2062</td>
</tr>
<tr>
<td>844</td>
<td>Data Format Structure for 3CMBT, G1, precipTotDm</td>
<td>2062</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>845</td>
<td>Data Format Structure for 3CMBT, G1, precipTotLogNw</td>
<td>2062</td>
</tr>
<tr>
<td>846</td>
<td>Data Format Structure for 3CMBT, G1, surfPrecipTotRateDiurnal</td>
<td>2062</td>
</tr>
<tr>
<td>847</td>
<td>Data Format Structure for 3CMBT, G2, precipTotRate</td>
<td>2063</td>
</tr>
<tr>
<td>848</td>
<td>Data Format Structure for 3CMBT, G2, precipLiqRate</td>
<td>2064</td>
</tr>
<tr>
<td>849</td>
<td>Data Format Structure for 3CMBT, G2, precipTotWaterContent</td>
<td>2064</td>
</tr>
<tr>
<td>850</td>
<td>Data Format Structure for 3CMBT, G2, precipLiqWaterContent</td>
<td>2064</td>
</tr>
<tr>
<td>851</td>
<td>Data Format Structure for 3CMBT, G2, cloudLiqWaterContent</td>
<td>2065</td>
</tr>
<tr>
<td>852</td>
<td>Data Format Structure for 3CMBT, G2, precipTotDm</td>
<td>2065</td>
</tr>
<tr>
<td>853</td>
<td>Data Format Structure for 3CMBT, G2, precipTotLogNw</td>
<td>2065</td>
</tr>
<tr>
<td>854</td>
<td>Data Format Structure for 3CMBT, G2, surfPrecipTotRateDiurnal</td>
<td>2065</td>
</tr>
<tr>
<td>855</td>
<td>Data Format Structure for 2AKuX, Ku precipitation</td>
<td>2084</td>
</tr>
<tr>
<td>856</td>
<td>Data Format Structure for 2AKuX, ScanTime</td>
<td>2085</td>
</tr>
<tr>
<td>857</td>
<td>Data Format Structure for 2AKuX, scanStatus</td>
<td>2085</td>
</tr>
<tr>
<td>858</td>
<td>Data Format Structure for 2AKuX, navigation</td>
<td>2086</td>
</tr>
<tr>
<td>859</td>
<td>Data Format Structure for 2AKuX, PRE</td>
<td>2087</td>
</tr>
<tr>
<td>860</td>
<td>Data Format Structure for 2AKuX, VER</td>
<td>2087</td>
</tr>
<tr>
<td>861</td>
<td>Data Format Structure for 2AKuX, CSF</td>
<td>2088</td>
</tr>
<tr>
<td>862</td>
<td>Data Format Structure for 2AKuX, SRT</td>
<td>2089</td>
</tr>
<tr>
<td>863</td>
<td>Data Format Structure for 2AKuX, DSD</td>
<td>2089</td>
</tr>
<tr>
<td>864</td>
<td>Data Format Structure for 2AKuX, Experimental</td>
<td>2089</td>
</tr>
<tr>
<td>865</td>
<td>Data Format Structure for 2AKuX, SLV</td>
<td>2090</td>
</tr>
<tr>
<td>866</td>
<td>Data Format Structure for 2AKuX, FLG</td>
<td>2091</td>
</tr>
<tr>
<td>867</td>
<td>Data Format Structure for 2AKaX, Ka precipitation</td>
<td>2130</td>
</tr>
<tr>
<td>868</td>
<td>Data Format Structure for 2AKaX, FS</td>
<td>2131</td>
</tr>
<tr>
<td>869</td>
<td>Data Format Structure for 2AKaX, HS</td>
<td>2131</td>
</tr>
<tr>
<td>870</td>
<td>Data Format Structure for 2AKaX, FS, ScanTime</td>
<td>2132</td>
</tr>
<tr>
<td>871</td>
<td>Data Format Structure for 2AKaX, FS, scanStatus</td>
<td>2132</td>
</tr>
<tr>
<td>872</td>
<td>Data Format Structure for 2AKaX, FS, navigation</td>
<td>2133</td>
</tr>
<tr>
<td>873</td>
<td>Data Format Structure for 2AKaX, FS, PRE</td>
<td>2134</td>
</tr>
<tr>
<td>874</td>
<td>Data Format Structure for 2AKaX, FS, VER</td>
<td>2134</td>
</tr>
<tr>
<td>875</td>
<td>Data Format Structure for 2AKaX, FS, CSF</td>
<td>2135</td>
</tr>
<tr>
<td>876</td>
<td>Data Format Structure for 2AKaX, FS, SRT</td>
<td>2136</td>
</tr>
<tr>
<td>877</td>
<td>Data Format Structure for 2AKaX, FS, DSD</td>
<td>2136</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>878</td>
<td>Data Format Structure for 2AKaX, FS, Experimental</td>
<td>2136</td>
</tr>
<tr>
<td>879</td>
<td>Data Format Structure for 2AKaX, FS, SLV</td>
<td>2137</td>
</tr>
<tr>
<td>880</td>
<td>Data Format Structure for 2AKaX, FS, FLG</td>
<td>2137</td>
</tr>
<tr>
<td>881</td>
<td>Data Format Structure for 2AKaX, HS, ScanTime</td>
<td>2138</td>
</tr>
<tr>
<td>882</td>
<td>Data Format Structure for 2AKaX, HS, scanStatus</td>
<td>2138</td>
</tr>
<tr>
<td>883</td>
<td>Data Format Structure for 2AKaX, HS, navigation</td>
<td>2139</td>
</tr>
<tr>
<td>884</td>
<td>Data Format Structure for 2AKaX, HS, PRE</td>
<td>2140</td>
</tr>
<tr>
<td>885</td>
<td>Data Format Structure for 2AKaX, HS, VER</td>
<td>2140</td>
</tr>
<tr>
<td>886</td>
<td>Data Format Structure for 2AKaX, HS, CSF</td>
<td>2141</td>
</tr>
<tr>
<td>887</td>
<td>Data Format Structure for 2AKaX, HS, SRT</td>
<td>2142</td>
</tr>
<tr>
<td>888</td>
<td>Data Format Structure for 2AKaX, HS, DSD</td>
<td>2142</td>
</tr>
<tr>
<td>889</td>
<td>Data Format Structure for 2AKaX, HS, Experimental</td>
<td>2143</td>
</tr>
<tr>
<td>890</td>
<td>Data Format Structure for 2AKaX, HS, SLV</td>
<td>2144</td>
</tr>
<tr>
<td>891</td>
<td>Data Format Structure for 2AKaX, HS, FLG</td>
<td>2144</td>
</tr>
<tr>
<td>892</td>
<td>Data Format Structure for 2ADPRX, DPR precipitation</td>
<td>2219</td>
</tr>
<tr>
<td>893</td>
<td>Data Format Structure for 2ADPRX, FS</td>
<td>2219</td>
</tr>
<tr>
<td>894</td>
<td>Data Format Structure for 2ADPRX, HS</td>
<td>2220</td>
</tr>
<tr>
<td>895</td>
<td>Data Format Structure for 2ADPRX, FS, ScanTime</td>
<td>2220</td>
</tr>
<tr>
<td>896</td>
<td>Data Format Structure for 2ADPRX, FS, scanStatus</td>
<td>2221</td>
</tr>
<tr>
<td>897</td>
<td>Data Format Structure for 2ADPRX, FS, navigation</td>
<td>2221</td>
</tr>
<tr>
<td>898</td>
<td>Data Format Structure for 2ADPRX, FS, PRE</td>
<td>2222</td>
</tr>
<tr>
<td>899</td>
<td>Data Format Structure for 2ADPRX, FS, VER</td>
<td>2222</td>
</tr>
<tr>
<td>900</td>
<td>Data Format Structure for 2ADPRX, FS, CSF</td>
<td>2223</td>
</tr>
<tr>
<td>901</td>
<td>Data Format Structure for 2ADPRX, FS, SRT</td>
<td>2224</td>
</tr>
<tr>
<td>902</td>
<td>Data Format Structure for 2ADPRX, FS, DSD</td>
<td>2224</td>
</tr>
<tr>
<td>903</td>
<td>Data Format Structure for 2ADPRX, FS, Experimental</td>
<td>2224</td>
</tr>
<tr>
<td>904</td>
<td>Data Format Structure for 2ADPRX, FS, SLV</td>
<td>2225</td>
</tr>
<tr>
<td>905</td>
<td>Data Format Structure for 2ADPRX, FS, FLG</td>
<td>2225</td>
</tr>
<tr>
<td>906</td>
<td>Data Format Structure for 2ADPRX, TRG</td>
<td>2226</td>
</tr>
<tr>
<td>907</td>
<td>Data Format Structure for 2ADPRX, FS, TRG</td>
<td>2227</td>
</tr>
<tr>
<td>908</td>
<td>Data Format Structure for 2ADPRX, HS, ScanTime</td>
<td>2227</td>
</tr>
<tr>
<td>909</td>
<td>Data Format Structure for 2ADPRX, HS, scanStatus</td>
<td>2227</td>
</tr>
<tr>
<td>910</td>
<td>Data Format Structure for 2ADPRX, HS, navigation</td>
<td>2228</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>911</td>
<td>Data Format Structure for 2ADPRX, HS, PRE</td>
<td>2229</td>
</tr>
<tr>
<td>912</td>
<td>Data Format Structure for 2ADPRX, HS, VER</td>
<td>2229</td>
</tr>
<tr>
<td>913</td>
<td>Data Format Structure for 2ADPRX, HS, CSF</td>
<td>2230</td>
</tr>
<tr>
<td>914</td>
<td>Data Format Structure for 2ADPRX, HS, SRT</td>
<td>2231</td>
</tr>
<tr>
<td>915</td>
<td>Data Format Structure for 2ADPRX, HS, DSD</td>
<td>2232</td>
</tr>
<tr>
<td>916</td>
<td>Data Format Structure for 2ADPRX, HS, Experimental</td>
<td>2232</td>
</tr>
<tr>
<td>917</td>
<td>Data Format Structure for 2ADPRX, HS, SLV</td>
<td>2233</td>
</tr>
<tr>
<td>918</td>
<td>Data Format Structure for 2ADPRX, HS, FLG</td>
<td>2233</td>
</tr>
<tr>
<td>919</td>
<td>Data Format Structure for 2AKuTMPX, Ku Temporary</td>
<td>2316</td>
</tr>
<tr>
<td>920</td>
<td>Data Format Structure for 2AKuTMPX, ScanTime</td>
<td>2316</td>
</tr>
<tr>
<td>921</td>
<td>Data Format Structure for 2AKuTMPX, scanStatus</td>
<td>2317</td>
</tr>
<tr>
<td>922</td>
<td>Data Format Structure for 2AKuTMPX, VertLocate</td>
<td>2318</td>
</tr>
<tr>
<td>923</td>
<td>Data Format Structure for 2AKuTMPX, Transmitter</td>
<td>2318</td>
</tr>
<tr>
<td>924</td>
<td>Data Format Structure for 2AKuTMPX, Receiver</td>
<td>2319</td>
</tr>
<tr>
<td>925</td>
<td>Data Format Structure for 2AKuTMPX, PRETMP</td>
<td>2319</td>
</tr>
<tr>
<td>926</td>
<td>Data Format Structure for 2AKuTMPX, VERTMP</td>
<td>2319</td>
</tr>
<tr>
<td>927</td>
<td>Data Format Structure for 2AKuTMPX, DSDTMP</td>
<td>2320</td>
</tr>
<tr>
<td>928</td>
<td>Data Format Structure for 2AKuTMPX, SLVTMP</td>
<td>2320</td>
</tr>
<tr>
<td>929</td>
<td>Data Format Structure for 2AKaTMPX, Ka Temporary</td>
<td>2340</td>
</tr>
<tr>
<td>930</td>
<td>Data Format Structure for 2AKaTMPX, FS</td>
<td>2340</td>
</tr>
<tr>
<td>931</td>
<td>Data Format Structure for 2AKaTMPX, HS</td>
<td>2341</td>
</tr>
<tr>
<td>932</td>
<td>Data Format Structure for 2AKaTMPX, FS, ScanTime</td>
<td>2341</td>
</tr>
<tr>
<td>933</td>
<td>Data Format Structure for 2AKaTMPX, FS, scanStatus</td>
<td>2342</td>
</tr>
<tr>
<td>934</td>
<td>Data Format Structure for 2AKaTMPX, FS, VertLocate</td>
<td>2343</td>
</tr>
<tr>
<td>935</td>
<td>Data Format Structure for 2AKaTMPX, FS, Transmitter</td>
<td>2343</td>
</tr>
<tr>
<td>936</td>
<td>Data Format Structure for 2AKaTMPX, FS, Receiver</td>
<td>2344</td>
</tr>
<tr>
<td>937</td>
<td>Data Format Structure for 2AKaTMPX, FS, PRETMP</td>
<td>2344</td>
</tr>
<tr>
<td>938</td>
<td>Data Format Structure for 2AKaTMPX, FS, VERTMP</td>
<td>2344</td>
</tr>
<tr>
<td>939</td>
<td>Data Format Structure for 2AKaTMPX, FS, DSDTMP</td>
<td>2345</td>
</tr>
<tr>
<td>940</td>
<td>Data Format Structure for 2AKaTMPX, FS, SLVTMP</td>
<td>2345</td>
</tr>
<tr>
<td>941</td>
<td>Data Format Structure for 2AKaTMPX, HS, ScanTime</td>
<td>2346</td>
</tr>
<tr>
<td>942</td>
<td>Data Format Structure for 2AKaTMPX, HS, scanStatus</td>
<td>2347</td>
</tr>
<tr>
<td>943</td>
<td>Data Format Structure for 2AKaTMPX, HS, VertLocate</td>
<td>2348</td>
</tr>
</tbody>
</table>
944 Data Format Structure for 2AKaTMPX, HS, Transmitter ............... 2348
945 Data Format Structure for 2AKaTMPX, HS, Receiver ..................... 2349
946 Data Format Structure for 2AKaTMPX, HS, PRETMP ....................... 2349
947 Data Format Structure for 2AKaTMPX, HS, VERTMP ....................... 2349
948 Data Format Structure for 2AKaTMPX, HS, DSDTMP ....................... 2350
949 Data Format Structure for 2AKaTMPX, HS, SLVTMP ....................... 2350
950 Data Format Structure for 2ADPRTMP, DPR Temporary ................... 2388
951 Data Format Structure for 2ADPRTMP, FS .............................. 2389
952 Data Format Structure for 2ADPRTMP, HS .............................. 2389
953 Data Format Structure for 2ADPRTMP, FS, ScanTime .................... 2389
954 Data Format Structure for 2ADPRTMP, FS, scanStatus ................... 2390
955 Data Format Structure for 2ADPRTMP, FS, PRETMP ....................... 2390
956 Data Format Structure for 2ADPRTMP, FS, VERTMP ....................... 2390
957 Data Format Structure for 2ADPRTMP, FS, DSDTMP ....................... 2391
958 Data Format Structure for 2ADPRTMP, FS, SLVTMP ....................... 2391
959 Data Format Structure for 2ADPRTMP, HS, ScanTime .................... 2391
960 Data Format Structure for 2ADPRTMP, HS, scanStatus .................... 2392
961 Data Format Structure for 2ADPRTMP, HS, PRETMP ....................... 2392
962 Data Format Structure for 2ADPRTMP, HS, VERTMP ....................... 2393
963 Data Format Structure for 2ADPRTMP, HS, DSDTMP ....................... 2393
964 Data Format Structure for 2ADPRTMP, HS, SLVTMP ....................... 2393
965 Data Format Structure for 2AKuENVX, Ku environment .................. 2418
966 Data Format Structure for 2AKuENVX, ScanTime ........................ 2419
967 Data Format Structure for 2AKuENVX, VERENV .......................... 2419
968 Data Format Structure for 2AKaENVX, Ka environment .................. 2425
969 Data Format Structure for 2AKaENVX, FS .............................. 2425
970 Data Format Structure for 2AKaENVX, HS .............................. 2425
971 Data Format Structure for 2AKaENVX, FS, ScanTime .................... 2426
972 Data Format Structure for 2AKaENVX, FS, VERENV ....................... 2426
973 Data Format Structure for 2AKaENVX, HS, ScanTime .................... 2426
974 Data Format Structure for 2AKaENVX, HS, VERENV ....................... 2427
975 Data Format Structure for 2ADPRENVX, DPR environment ................ 2436
976 Data Format Structure for 2ADPRENVX, FS .............................. 2436
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>977</td>
<td>Data Format Structure for 2ADPENVX, HS</td>
<td>2436</td>
</tr>
<tr>
<td>978</td>
<td>Data Format Structure for 2ADPENVX, FS, ScanTime</td>
<td>2437</td>
</tr>
<tr>
<td>979</td>
<td>Data Format Structure for 2ADPENVX, FS, VERENV</td>
<td>2437</td>
</tr>
<tr>
<td>980</td>
<td>Data Format Structure for 2ADPENVX, HS, ScanTime</td>
<td>2437</td>
</tr>
<tr>
<td>981</td>
<td>Data Format Structure for 2ADPENVX, HS, VERENV</td>
<td>2438</td>
</tr>
<tr>
<td>982</td>
<td>Data Format Structure for 3DPRX, DPR Full Product</td>
<td>2449</td>
</tr>
<tr>
<td>983</td>
<td>Data Format Structure for 3DPRX, G1, G1</td>
<td>2450</td>
</tr>
<tr>
<td>984</td>
<td>Data Format Structure for 3DPRX, G1, G1</td>
<td>2451</td>
</tr>
<tr>
<td>985</td>
<td>Data Format Structure for 3DPRX, G1</td>
<td>2452</td>
</tr>
<tr>
<td>986</td>
<td>Data Format Structure for 3DPRX, G2, G2</td>
<td>2452</td>
</tr>
<tr>
<td>987</td>
<td>Data Format Structure for 3DPRX, G2</td>
<td>2453</td>
</tr>
<tr>
<td>988</td>
<td>Data Format Structure for 3DPRX, G1, precipRate</td>
<td>2453</td>
</tr>
<tr>
<td>989</td>
<td>Data Format Structure for 3DPRX, G1, rainRate</td>
<td>2454</td>
</tr>
<tr>
<td>990</td>
<td>Data Format Structure for 3DPRX, G1, snowRate</td>
<td>2454</td>
</tr>
<tr>
<td>991</td>
<td>Data Format Structure for 3DPRX, G1, flagHeavyIcePrecip</td>
<td>2454</td>
</tr>
<tr>
<td>992</td>
<td>Data Format Structure for 3DPRX, G1, mixedPhRate</td>
<td>2454</td>
</tr>
<tr>
<td>993</td>
<td>Data Format Structure for 3DPRX, G1, precipRateESurface</td>
<td>2454</td>
</tr>
<tr>
<td>994</td>
<td>Data Format Structure for 3DPRX, G1, precipRateESurface2</td>
<td>2455</td>
</tr>
<tr>
<td>995</td>
<td>Data Format Structure for 3DPRX, G1, precipRateNearSurface</td>
<td>2455</td>
</tr>
<tr>
<td>996</td>
<td>Data Format Structure for 3DPRX, G1, rainRateNearSurface</td>
<td>2455</td>
</tr>
<tr>
<td>997</td>
<td>Data Format Structure for 3DPRX, G1, snowRateNearSurface</td>
<td>2456</td>
</tr>
<tr>
<td>998</td>
<td>Data Format Structure for 3DPRX, G1, mixedPhRateNearSurface</td>
<td>2456</td>
</tr>
<tr>
<td>999</td>
<td>Data Format Structure for 3DPRX, G1, precipWaterIntegrated</td>
<td>2456</td>
</tr>
<tr>
<td>1000</td>
<td>Data Format Structure for 3DPRX, G1, precipIceIntegrated</td>
<td>2456</td>
</tr>
<tr>
<td>1001</td>
<td>Data Format Structure for 3DPRX, G1, precipRateAve24</td>
<td>2456</td>
</tr>
<tr>
<td>1002</td>
<td>Data Format Structure for 3DPRX, G1, zFactorCorrected</td>
<td>2457</td>
</tr>
<tr>
<td>1003</td>
<td>Data Format Structure for 3DPRX, G1, zFactorCorrectedESurface</td>
<td>2457</td>
</tr>
<tr>
<td>1004</td>
<td>Data Format Structure for 3DPRX, G1, zFactorCorrectedNearSurface</td>
<td>2457</td>
</tr>
<tr>
<td>1005</td>
<td>Data Format Structure for 3DPRX, G1, zFactorMeasuredNearSurface</td>
<td>2457</td>
</tr>
<tr>
<td>1006</td>
<td>Data Format Structure for 3DPRX, G1, zFactorCorrectedDPR</td>
<td>2457</td>
</tr>
<tr>
<td>1007</td>
<td>Data Format Structure for 3DPRX, G1, zFactorCorrectedESurfaceDPR</td>
<td>2458</td>
</tr>
<tr>
<td>1008</td>
<td>Data Format Structure for 3DPRX, G1, zFactorCorrectedNearSurfaceDPR</td>
<td>2458</td>
</tr>
<tr>
<td>1009</td>
<td>Data Format Structure for 3DPRX, G1, zFactorMeasured</td>
<td>2458</td>
</tr>
</tbody>
</table>
1010 Data Format Structure for 3DPRX, G1, dm ........................... 2458
1011 Data Format Structure for 3DPRX, G1, dBNW ........................ 2458
1012 Data Format Structure for 3DPRX, G1, epsilonDPR ...................... 2459
1013 Data Format Structure for 3DPRX, G1, epsilon ................................ 2459
1014 Data Format Structure for 3DPRX, G1, zeta ................................ 2459
1015 Data Format Structure for 3DPRX, G1, piaHB ................................ 2459
1016 Data Format Structure for 3DPRX, G1, piaHybrid .......................... 2459
1017 Data Format Structure for 3DPRX, G1, piaHybridDPR ..................... 2460
1018 Data Format Structure for 3DPRX, G1, piaSRT ............................... 2460
1019 Data Format Structure for 3DPRX, G1, piaSRTdpr .......................... 2460
1020 Data Format Structure for 3DPRX, G1, piaFinal ............................ 2460
1021 Data Format Structure for 3DPRX, G1, piaFinalDPR ....................... 2460
1022 Data Format Structure for 3DPRX, G1, piaFinalSubset ..................... 2461
1023 Data Format Structure for 3DPRX, G1, piaFinalDPRsubset ............... 2461
1024 Data Format Structure for 3DPRX, G1, heightBB .......................... 2461
1025 Data Format Structure for 3DPRX, G1, heightBNNadir .................... 2461
1026 Data Format Structure for 3DPRX, G1, BBwidthNadir ..................... 2461
1027 Data Format Structure for 3DPRX, G1, heightStormTop .................... 2462
1028 Data Format Structure for 3DPRX, G1, BBwidth ............................ 2462
1029 Data Format Structure for 3DPRX, G1, observationCounts ................ 2462
1030 Data Format Structure for 3DPRX, G1, precipRateLocalTime ............. 2462
1031 Data Format Structure for 3DPRX, G1, DFRmNearSurface .................. 2462
1032 Data Format Structure for 3DPRX, G1, DFRNearSurface .................... 2462
1033 Data Format Structure for 3DPRX, G2, precipRate .......................... 2463
1034 Data Format Structure for 3DPRX, G2, rainRate ............................. 2463
1035 Data Format Structure for 3DPRX, G2, snowRate ............................. 2463
1036 Data Format Structure for 3DPRX, G2, flagHeavyIcePrecip ............... 2463
1037 Data Format Structure for 3DPRX, G2, mixedPhRate ........................ 2464
1038 Data Format Structure for 3DPRX, G2, precipRateESurface ............... 2464
1039 Data Format Structure for 3DPRX, G2, precipRateESurface2 ............... 2464
1040 Data Format Structure for 3DPRX, G2, precipRateNearSurface ............. 2464
1041 Data Format Structure for 3DPRX, G2, rainRateNearSurface ................ 2464
1042 Data Format Structure for 3DPRX, G2, snowRateNearSurface ............... 2464
LIST OF FIGURES

1043 Data Format Structure for 3DPRX, G2, mixedPhRateNearSurface . . . . . . . 2464
1044 Data Format Structure for 3DPRX, G2, precipWaterIntegrated . . . . . . . 2465
1045 Data Format Structure for 3DPRX, G2, precipIceIntegrated . . . . . . . . . 2465
1046 Data Format Structure for 3DPRX, G2, precipRateAve24 . . . . . . . . . . . 2465
1047 Data Format Structure for 3DPRX, G2, zFactorCorrected . . . . . . . . . . . 2466
1048 Data Format Structure for 3DPRX, G2, zFactorCorrectedESurface . . . . . . 2466
1049 Data Format Structure for 3DPRX, G2, zFactorCorrectedNearSurface . . . 2466
1050 Data Format Structure for 3DPRX, G2, zFactorMeasuredNearSurface . . . . 2466
1051 Data Format Structure for 3DPRX, G2, zFactorCorrectedDPR . . . . . . . . . 2466
1052 Data Format Structure for 3DPRX, G2, zFactorCorrectedESurfaceDPR . . . 2466
1053 Data Format Structure for 3DPRX, G2, zFactorCorrectedNearSurfaceDPR . 2466
1054 Data Format Structure for 3DPRX, G2, zFactorMeasured . . . . . . . . . . . 2467
1055 Data Format Structure for 3DPRX, G2, dm . . . . . . . . . . . . . . . . . . . 2467
1056 Data Format Structure for 3DPRX, G2, dBNw . . . . . . . . . . . . . . . . . 2467
1057 Data Format Structure for 3DPRX, G2, epsilonDPR . . . . . . . . . . . . . . 2468
1058 Data Format Structure for 3DPRX, G2, epsilon . . . . . . . . . . . . . . . . 2468
1059 Data Format Structure for 3DPRX, G2, zeta . . . . . . . . . . . . . . . . . . 2468
1060 Data Format Structure for 3DPRX, G2, piaHB . . . . . . . . . . . . . . . . 2468
1061 Data Format Structure for 3DPRX, G2, piaHybrid . . . . . . . . . . . . . . 2468
1062 Data Format Structure for 3DPRX, G2, piaHybridDPR . . . . . . . . . . . . 2468
1063 Data Format Structure for 3DPRX, G2, piaSRT . . . . . . . . . . . . . . . . 2468
1064 Data Format Structure for 3DPRX, G2, piaSRTdpr . . . . . . . . . . . . . . 2469
1065 Data Format Structure for 3DPRX, G2, piaFinal . . . . . . . . . . . . . . . 2469
1066 Data Format Structure for 3DPRX, G2, piaFinalDPR . . . . . . . . . . . . . 2469
1067 Data Format Structure for 3DPRX, G2, heightBB . . . . . . . . . . . . . . . 2470
1068 Data Format Structure for 3DPRX, G2, heightStormTop . . . . . . . . . . 2470
1069 Data Format Structure for 3DPRX, G2, BBwidth . . . . . . . . . . . . . . . 2470
1070 Data Format Structure for 3DPRX, G2, observationCounts . . . . . . . . . 2471
1071 Data Format Structure for 3DPRX, G2, DFRmNearSurface . . . . . . . . . 2471
1072 Data Format Structure for 3DPRX, G2, DFRNearSurface . . . . . . . . . . 2471
1073 Data Format Structure for 2BCMBX, Level-2 DPR and GMI Combined . . 2546
1074 Data Format Structure for 2BCMBX, NS, NS . . . . . . . . . . . . . . . . . . 2547
1075 Data Format Structure for 2BCMBX, NS . . . . . . . . . . . . . . . . . . . 2548
LIST OF FIGURES

1076 Data Format Structure for 2BCMBX, FS, FS .......................... 2549
1077 Data Format Structure for 2BCMBX, FS ............................... 2550
1078 Data Format Structure for 2BCMBX, NS, ScanTime .................. 2551
1079 Data Format Structure for 2BCMBX, NS, scanStatus .................. 2551
1080 Data Format Structure for 2BCMBX, NS, navigation ................... 2552
1081 Data Format Structure for 2BCMBX, NS, Input ........................ 2553
1082 Data Format Structure for 2BCMBX, NS, aPriori ....................... 2553
1083 Data Format Structure for 2BCMBX, NS, FLG .......................... 2553
1084 Data Format Structure for 2BCMBX, FS, ScanTime .................... 2554
1085 Data Format Structure for 2BCMBX, FS, scanStatus ................... 2554
1086 Data Format Structure for 2BCMBX, FS, navigation ................... 2555
1087 Data Format Structure for 2BCMBX, FS, Input ........................ 2556
1088 Data Format Structure for 2BCMBX, FS, aPriori ....................... 2557
1089 Data Format Structure for 2BCMBX, FS, FLG .......................... 2557
1090 Data Format Structure for 3CMBX, Combined precipitation .............. 2600
1091 Data Format Structure for 3CMBX, G1 .................................. 2601
1092 Data Format Structure for 3CMBX, G2 .................................. 2602
1093 Data Format Structure for 3CMBX, G1, precipTotRate .................. 2602
1094 Data Format Structure for 3CMBX, G1, precipLiqRate .................. 2602
1095 Data Format Structure for 3CMBX, G1, precipTotWaterContent ........... 2603
1096 Data Format Structure for 3CMBX, G1, precipLiqWaterContent ........... 2603
1097 Data Format Structure for 3CMBX, G1, cloudLiqWaterContent ............ 2603
1098 Data Format Structure for 3CMBX, G1, precipTotDm ..................... 2603
1099 Data Format Structure for 3CMBX, G1, precipTotLogNw ................ 2603
1100 Data Format Structure for 3CMBX, G1, surfPrecipTotRateDiurnal .......... 2603
1101 Data Format Structure for 3CMBX, G2, precipTotRate .................. 2604
1102 Data Format Structure for 3CMBX, G2, precipLiqRate .................. 2605
1103 Data Format Structure for 3CMBX, G2, precipTotWaterContent ........... 2605
1104 Data Format Structure for 3CMBX, G2, precipLiqWaterContent ........... 2605
1105 Data Format Structure for 3CMBX, G2, cloudLiqWaterContent ............ 2606
1106 Data Format Structure for 3CMBX, G2, precipTotDm ..................... 2606
1107 Data Format Structure for 3CMBX, G2, precipTotLogNw ................ 2606
1108 Data Format Structure for 3CMBX, G2, surfPrecipTotRateDiurnal .......... 2606
<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1109</td>
<td>Data Format Structure for 3CMBTX, Combined precipitation</td>
<td>2625</td>
</tr>
<tr>
<td>1110</td>
<td>Data Format Structure for 3CMBTX, G1</td>
<td>2625</td>
</tr>
<tr>
<td>1111</td>
<td>Data Format Structure for 3CMBTX, G2</td>
<td>2626</td>
</tr>
<tr>
<td>1112</td>
<td>Data Format Structure for 3CMBTX, G1, precipTotRate</td>
<td>2626</td>
</tr>
<tr>
<td>1113</td>
<td>Data Format Structure for 3CMBTX, G1, precipLiqRate</td>
<td>2626</td>
</tr>
<tr>
<td>1114</td>
<td>Data Format Structure for 3CMBTX, G1, precipTotWaterContent</td>
<td>2627</td>
</tr>
<tr>
<td>1115</td>
<td>Data Format Structure for 3CMBTX, G1, precipLiqWaterContent</td>
<td>2627</td>
</tr>
<tr>
<td>1116</td>
<td>Data Format Structure for 3CMBTX, G1, cloudLiqWaterContent</td>
<td>2627</td>
</tr>
<tr>
<td>1117</td>
<td>Data Format Structure for 3CMBTX, G1, precipTotDm</td>
<td>2627</td>
</tr>
<tr>
<td>1118</td>
<td>Data Format Structure for 3CMBTX, G1, precipTotLogNw</td>
<td>2627</td>
</tr>
<tr>
<td>1119</td>
<td>Data Format Structure for 3CMBTX, G1, surfPrecipTotRateDiurnal</td>
<td>2627</td>
</tr>
<tr>
<td>1120</td>
<td>Data Format Structure for 3CMBTX, G2, precipTotRate</td>
<td>2628</td>
</tr>
<tr>
<td>1121</td>
<td>Data Format Structure for 3CMBTX, G2, precipLiqRate</td>
<td>2629</td>
</tr>
<tr>
<td>1122</td>
<td>Data Format Structure for 3CMBTX, G2, precipTotWaterContent</td>
<td>2629</td>
</tr>
<tr>
<td>1123</td>
<td>Data Format Structure for 3CMBTX, G2, precipLiqWaterContent</td>
<td>2629</td>
</tr>
<tr>
<td>1124</td>
<td>Data Format Structure for 3CMBTX, G2, cloudLiqWaterContent</td>
<td>2630</td>
</tr>
<tr>
<td>1125</td>
<td>Data Format Structure for 3CMBTX, G2, precipTotDm</td>
<td>2630</td>
</tr>
<tr>
<td>1126</td>
<td>Data Format Structure for 3CMBTX, G2, precipTotLogNw</td>
<td>2630</td>
</tr>
<tr>
<td>1127</td>
<td>Data Format Structure for 3CMBTX, G2, surfPrecipTotRateDiurnal</td>
<td>2630</td>
</tr>
<tr>
<td>1128</td>
<td>Data Format Structure for 3GSMAPH4, GSMaP Hourly</td>
<td>2649</td>
</tr>
<tr>
<td>1129</td>
<td>Data Format Structure for 3GSMAPM4, GSMaP Monthly</td>
<td>2654</td>
</tr>
<tr>
<td>1130</td>
<td>Data Format Structure for 3IMERGHH, IMERG 30-minute</td>
<td>2657</td>
</tr>
<tr>
<td>1131</td>
<td>Data Format Structure for 3IMERGM, IMERG monthly</td>
<td>2662</td>
</tr>
<tr>
<td>1132</td>
<td>Data Format Structure for 2HSLH, Spectral Latent Heating</td>
<td>2666</td>
</tr>
<tr>
<td>1133</td>
<td>Data Format Structure for 2HSLH, Spectral Latent Heating</td>
<td>2667</td>
</tr>
<tr>
<td>1134</td>
<td>Data Format Structure for 2HSLH, ScanTime</td>
<td>2667</td>
</tr>
<tr>
<td>1135</td>
<td>Data Format Structure for 3GSLH, Gridded Orbital Spectral Latent Heating</td>
<td>2674</td>
</tr>
<tr>
<td>1136</td>
<td>Data Format Structure for 3GSLH, Gridded Orbital Spectral Latent Heating</td>
<td>2675</td>
</tr>
<tr>
<td>1137</td>
<td>Data Format Structure for 3GSLH, GridTime</td>
<td>2675</td>
</tr>
<tr>
<td>1138</td>
<td>Data Format Structure for 3HSLH, Monthly Spectral Latent Heating</td>
<td>2682</td>
</tr>
<tr>
<td>1139</td>
<td>Data Format Structure for 3HSLH, Monthly Spectral Latent Heating</td>
<td>2683</td>
</tr>
<tr>
<td>1140</td>
<td>Data Format Structure for 3HSLH, Monthly Spectral Latent Heating</td>
<td>2684</td>
</tr>
<tr>
<td>1141</td>
<td>Data Format Structure for 2HSLHT, Spectral Latent Heating</td>
<td>2692</td>
</tr>
</tbody>
</table>
1142 Data Format Structure for 2HSLHT, Spectral Latent Heating ........................................... 2693
1143 Data Format Structure for 2HSLHT, ScanTime ................................................................. 2693
1144 Data Format Structure for 3GSLHT, Gridded Orbital Spectral Latent Heating.................. 2700
1145 Data Format Structure for 3GSLHT, Gridded Orbital Spectral Latent Heating.................. 2701
1146 Data Format Structure for 3GSLHT, GridTime ................................................................. 2701
1147 Data Format Structure for 3HSLHT, Monthly Spectral Latent Heating ............................ 2708
1148 Data Format Structure for 3HSLHT, Monthly Spectral Latent Heating ............................ 2709
1149 Data Format Structure for 3HSLHT, Monthly Spectral Latent Heating ............................ 2710
1150 Data Format Structure for 2HCSH, Convective Stratiform Heating ................................. 2718
1151 Data Format Structure for 2HCSH, ScanTime ................................................................. 2718
1152 Data Format Structure for 3GCSH, Gridded Orbital Convective Stratiform Heating from Com imbined............................................................... 2728
1153 Data Format Structure for 3GCSH, GridTime ................................................................. 2723
1154 Data Format Structure for 3HCSH, Monthly Convective Stratiform Heating from Combined ............................................................... 2728
1155 Data Format Structure for 2HCSHT, Convective Stratiform Heating ............................... 2731
1156 Data Format Structure for 2HCSHT, ScanTime ............................................................... 2732
1157 Data Format Structure for 3GCSHT, Gridded Orbital Convective Stratiform Heating from Combined ............................................................... 2737
1158 Data Format Structure for 3GCSHT, GridTime ............................................................... 2737
1159 Data Format Structure for 3HCSHT, Monthly Convective Stratiform Heating from Combined ............................................................... 2741
1 Introduction

Global Precipitation Measurement (GPM) is an international satellite mission to provide observations of rain and snow worldwide. NASA and Japan Aerospace Exploration Agency (JAXA) launched the GPM Core satellite from Japan on February 27, 2014 UTC (February 28 Japan Standard Time). The data provided by the Core unifies precipitation measurements made by an international network of partner satellites to quantify when, where, and how much it rains or snows around the world.

1.1 Identification

This is the File Specification for GPM Products.

1.2 Scope

This document describes the data file formats for GPM products. Metadata is described in Metadata for GPM Products.

1.3 Purpose and Objectives

The purpose of this file specification document is to define the file content and format for the GPM data products.

1.4 Document Status and Schedule

The file specifications have been reviewed by the algorithm developers. Formats are expected to change for each processing cycle.

1.5 Document Organization

The organization is as follows:
Section 2 LOGICAL FORMAT - This section describes general aspects of the logical format.
Section 3 PHYSICAL FORMAT - This section describes general aspects of the physical format.
Section 4 FORMATTING CONVENTIONS - This section describes the general formatting conventions used in this document.
Section 5 STANDARD GPM PRODUCTS - This section describes the file specifications for the standard GPM products.
2 Logical Format

The logical format of a data product consists of the names, types, dimensions, and organization of the data. The physical format is the implementation of the logical format with an underlying format such as Hierarchical Data Format (HDF). The bulk of this document consists of the logical format of each GPM data product.

GPM data products contain metadata and data. Metadata are small text strings containing label information such as the name, date, and time of the data products. Metadata are often organized into metadata groups.

Data are arrays or scalers. Data are often organized into swath structures or grid structures. Some products have groups outside or inside swath structures or grid structures.

2.1 Swath Structure

The swath structure stores satellite data which are organized by scans. Swath structures are implemented in Levels 1A, 1B, 1C, 2A, and 2B. The swath structure is contained in a group. In this swath group is the metadata group SwathHeader, data group ScanTime, data arrays Latitude and Longitude and other data arrays. In some products there are additional data groups under the swath group. The contents of the metadata group SwathHeader are explained in Metadata for GPM Products.

2.2 Grid Structure

The grid structure stores earth located grids. Each grid is an array of grid boxes, rather than grid points. Grid structures are implemented in Level 3A and 3B products. The grid structure is contained in a grid group. In this group is the metadata group GridHeader and data arrays. In some products there are additional data groups under the grid group. The contents of the metadata group GridHeader are explained in Metadata for GPM Products.

3 Physical Format

The logical format of GPM data products is written in an underlying format such as HDF.

3.1 Hierarchical Data Format

HDF was developed by the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Champaign-Urbana and is the archive format for GPM data. HDF manuals and software may be obtained via anonymous ftp at ftp.ncsa.uiuc.edu
The logical group is implemented in HDF as a Vgroup. The logical array or scalar is implemented in HDF as a Scientific Data Set. Each SDS contains the data array and additional information as attributes: names of dimensions, units, scale, offset, and scale description.

Each metadata group is implemented in HDF as an attribute. Elements within a group are implemented as

```
ElementName=ElementValue;
```

If the element has a list of values, the values are separated with a comma:

```
ElementName=Value1,Value2,...,ValueN;
```

## 4 Formatting Conventions

### 4.1 File Structure Figure

Each data product section has a file structure figure and file contents. The file structure figure show the organization of the data within the file. The File is on the left. Under the file are circles showing swaths or grid structures or boxes showing metadata, groups, or arrays. Group boxes are shaded. Array boxes contain the size of one element with the dimensions to the right of the box. A group has an additional figure showing the contents of the group.

### 4.2 File Contents

Each array or scaler is described with name in bold, then parenthesis containing the data type and dimensions, and then a description.

### 4.3 Missing Data and Empty Granules

Missing data are denoted by values equal to -9999.9, -9999.9, -9999, -9999, -99, 65535, 4294967295, 255, and NULL for for 8-byte float, 4-byte float, 8-byte integer, 4-byte integer, 2-byte integer, 1-byte integer, 2-byte unsigned integer, 4-byte unsigned integer, 1-byte character, and variable length string. Any exceptions to the use of the above standard values are explicitly notes in the description.

If an entire granule is missing, an empty granule may be created. An empty granule is defined by the metadata element EmptyGranule in the metadata group FileHeader. Software reading a granule should check EmptyGranule first. Swath data or grid data may be empty.
4.4 Array Dimension Order

In the definition of array dimensions, e.g. npixel x nscan, the first dimension (npixel) is the most rapidly varying index and the last dimension (nscan) is the least rapidly varying index. To implement the format in FORTRAN, declare an array with the dimensions as they appear in this document. To implement the format in C, declare an array with the dimensions reversed from their appearance in this document.

4.5 Array Index

The meaning of each array index is explained at the beginning of each algorithm section in the list called "Dimension Definitions." Some array indices denote a type rather than a number. For example, PIAalt has dimensions method x nray x nscan and there are 6 methods. If an index is enumerated the index value will start with 1 (rather than 0) unless otherwise indicated.

4.6 Granule definition

For orbital products, the beginning and ending time are defined as the time the sub-satellite track reaches its southernmost latitude. A scan is included in a granule when its ScanTime is greater than or equal to the Granule start time and less than the Granule end time.

For time-averaged products, the beginning time is the first millisecond of the period and the ending time is the last millisecond.
5 Standard GPM Products

5.1 1AGMI - GMI unpacked packet data

1AGMI contains unpacked packet data from GMI science data from the GMI passive microwave instrument flown on the GPM satellite. Swath S1 has 9 channels which are similar to TRMM TMI (10V 10H 19V 19H 23V 37V 37H 89V 89H). Swath S2 has 4 channels similar to AMSU-B (166V 166H 183+/-3V 183+/-8V). Data for both swaths is observed in the same revolution of the instrument. Swath S3 has ScienceDataHeader. Swath S4 has full rotation for low freq channels (S1). Swath S5 has full rotation for high freq channels (S2).

GMI sample counts.  

The S1 channels are:

10.7 GHz vertically-polarized  
10.7 GHz horizontally-polarized  
18.7 GHz vertically-polarized  
18.7 GHz horizontally-polarized  
23.8 GHz vertically-polarized  
36.5 GHz vertically-polarized  
36.5 GHz horizontally-polarized  
89.0 GHz vertically-polarized  
89.0 GHz horizontally-polarized

GMI sample counts.  

The S2 channels are:

166.0 GHz vertically-polarized  
166.0 GHz horizontally-polarized  
183.31+/-3 GHz vertically-polarized  
183.31+/-8 GHz vertically-polarized

Earth observations are taken during a segment of the rotation when GMI is looking in the +x direction of the GPM satellite. Since the spacecraft turns around every few weeks, +x may be forward or aft. We define the spacecraft axis v, used in the definition of the variable SCorientation, at the center of this segment and the same as the +x direction.

32rpm * 1min/60s * 5538s/orbit = 2954 scans / orbit.

RELATION BETWEEN THE SWATHS: Swath S2 has the same number of scans and the same number of pixels as Swath S1. Each S1 scan contains 9 channels sampled 221 times along the scan. Each S2 scan contains 4 channels sampled 221 times along the scan. Since the incidence angle of Swath S1 is different than Swath S2, the geolocations of the pixel centers are different.
Dimension definitions:

VH    2    Number of polarizations.
nscan1  var  Typical number of Swath S1 scans in the granule.
nchannel1  9    Number of Swath S1 channels (10V 10H 19V 19H 23V 37V 37H
89V 89H).
npixelev 221    Number of earth view pixels in one scan.
npixelht 221    Number of hot load pixels in one scan.
npixelcs 221    Number of cold sky pixels in one scan.
nscan2  var  Typical number of Swath S2 scans in the granule.
nchannel2  4    Number of Swath S2 channels (166V 166H 183+/−3V 183+/−8V).
npixel3  1    Number of “pixels” in one scan in S3.
npixelfr 500    Number of full rotation earth view pixels in one scan.
nchannel12  13    Number of Swath S1 and S2 channels.
dim2  2    Number.
dim3  3    Number.
dim4  4    Number.
dim5  5    Number.
dim6  6    Number.
dim7  7    Number.
dim8  8    Number.
dim9  9    Number.
dim10  10    Number.
dim11  11    Number.
dim12  12    Number.
GMIxyz  3    x, y, z components in GMI instrument coordinate system.

Figure 1 through Figure 47 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
5.1 1AGMI - GMI unpacked packet data

Figure 1: Data Format Structure for 1AGMI, GMI unpacked packet data
Figure 2: Data Format Structure for 1AGMI, S1
5.1 1AGMI - GMI unpacked packet data

Figure 3: Data Format Structure for 1AGMI, S2
Figure 4: Data Format Structure for 1AGMI, S3, S3
5.1 1AGMI - GMI unpacked packet data

continued from last figure

<table>
<thead>
<tr>
<th>Group: nscan2</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEHSK_TEMP</td>
</tr>
<tr>
<td>IE_TELEMETRY</td>
</tr>
<tr>
<td>MECHANISMS</td>
</tr>
<tr>
<td>SMPL_INFO</td>
</tr>
</tbody>
</table>

Figure 5: Data Format Structure for 1AGMI, S3

<table>
<thead>
<tr>
<th>Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td>S4_SwathHeader</td>
</tr>
<tr>
<td>ScanTime</td>
</tr>
<tr>
<td>ephemerisUsed</td>
</tr>
<tr>
<td>Latitude</td>
</tr>
<tr>
<td>Longitude</td>
</tr>
<tr>
<td>incidenceAngle</td>
</tr>
<tr>
<td>fullRotation</td>
</tr>
<tr>
<td>fullRotBlanking</td>
</tr>
</tbody>
</table>

Figure 6: Data Format Structure for 1AGMI, S4

<table>
<thead>
<tr>
<th>Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td>S5_SwathHeader</td>
</tr>
<tr>
<td>ScanTime</td>
</tr>
<tr>
<td>ephemerisUsed</td>
</tr>
<tr>
<td>Latitude</td>
</tr>
<tr>
<td>Longitude</td>
</tr>
<tr>
<td>incidenceAngle</td>
</tr>
<tr>
<td>fullRotation</td>
</tr>
</tbody>
</table>

Figure 7: Data Format Structure for 1AGMI, S5

<table>
<thead>
<tr>
<th>Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td>gmi1aHeader</td>
</tr>
<tr>
<td>sampleRangeFile</td>
</tr>
</tbody>
</table>

Figure 8: Data Format Structure for 1AGMI, gmi1aHeader
Figure 9: Data Format Structure for 1AGMI, S1, ScanTime

Figure 10: Data Format Structure for 1AGMI, S1, scanStatus
5.1 1AGMI - GMI unpacked packet data

![Diagram of data format structure for 1AGMI, S1, navigation](image1)

Figure 11: Data Format Structure for 1AGMI, S1, navigation

![Diagram of data format structure for 1AGMI, S1, sunData](image2)

Figure 12: Data Format Structure for 1AGMI, S1, sunData
54

5 STANDARD GPM PRODUCTS

Figure 13: Data Format Structure for 1AGMI, S2, ScanTime

Figure 14: Data Format Structure for 1AGMI, S2, scanStatus
5.1 1AGMI - GMI unpacked packet data

Figure 15: Data Format Structure for 1AGMI, S2, navigation

Figure 16: Data Format Structure for 1AGMI, S2, sunData
Figure 17: Data Format Structure for 1AGMI, S3, ScanTime

Figure 18: Data Format Structure for 1AGMI, S3, TAM1

Figure 19: Data Format Structure for 1AGMI, S3, TAM2

Figure 20: Data Format Structure for 1AGMI, S3, TORQUE_BAR
5.1 1AGMI - GMI unpacked packet data

![Diagram of GMI_temperatures]

Figure 21: Data Format Structure for 1AGMI, S3, GMI_temperatures

![Diagram of primary-header]

Figure 22: Data Format Structure for 1AGMI, S3, primaryHeader

![Diagram of gsdr_time]

Figure 23: Data Format Structure for 1AGMI, S3, GSDR_TIME
Figure 24: Data Format Structure for 1AGMI, SENSOR_INFO

continued on next figure

•

•

Figure 24: Data Format Structure for 1AGMI, SENSOR_INFO
5.1 1AGMI - GMI unpacked packet data

continued from last figure

Figure 25: Data Format Structure for 1AGMI, S3, SENSOR_INFO
continued on next figure

Figure 26: Data Format Structure for 1AGMI, RS_INFO
### 5.1 1AGMI - GMI unpacked packet data

<table>
<thead>
<tr>
<th>Field</th>
<th>Size</th>
<th>Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLK_DELAY</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>BLK_DURATION</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>RS_HSK_SIZE</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>RS_PAR_ERR_CNT</td>
<td>4 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>RS_SCAN_CNT</td>
<td>4 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>SAMPLE_TBL_VER</td>
<td>4 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>SMPL_TBL</td>
<td>4 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>RS_SC_SIZE</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>GSDR_SIZE</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>GSDR_LEFT</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>GSDR_B_POP_IDX</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>GSDR_B_PUSH_IDX</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>GSDR_APID</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>PKT_STATE</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>OVRD_RS_PWR</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>OVRD_SMA_SPIN</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
<tr>
<td>OVRD_PASS_RS</td>
<td>2 bytes</td>
<td>Array: nscan2</td>
</tr>
</tbody>
</table>

Figure 27: Data Format Structure for 1AGMI, S3, RS_INFO
Figure 28: Data Format Structure for 1AGMI, SYNCH_STAMPS
5.1 1AGMI - GMI unpacked packet data

continued from last figure

Figure 29: Data Format Structure for 1AGMI, S3, SYNCH_STAMPS
Figure 30: Data Format Structure for 1AGMI, SYNCH_STAMPS2
Figure 31: Data Format Structure for 1AGMI, S3, SYNCH, STAMPS2
Figure 32: Data Format Structure for 1AGMI, RSHSK_STATUS
5.1 1AGMI - GMI unpacked packet data

Figure 33: Data Format Structure for 1AGMI, S3, RSHSK_STATUS

Figure 34: Data Format Structure for 1AGMI, S3, RSHSK_SAMPL_INFO
Figure 35: Data Format Structure for 1AGMI, S3, RSHSK_GAIN
Figure 36: Data Format Structure for 1AGMI, RSHSK_TEMP
Figure 37: Data Format Structure for 1AGMI, S3, RSHSK_TEMP
Figure 38: Data Format Structure for 1AGMI, IEHSK, TEMP
Figure 39: Data Format Structure for 1AGMI, S3, IEHSK_TEMP
5.1 1AGMI - GMI unpacked packet data

Figure 40: Data Format Structure for 1AGMI, IE_TELEMETRY

IE_TELEMETRY

- IE_PASSTHRU_RSP 4 bytes Array: nscan2
- IE_BLANKING_CNT 2 bytes Array: nscan2
- PWR_STAT_LR_PR 2 bytes Array: nscan2
- PWR_STAT_A 2 bytes Array: nscan2
- HTR_STAT_SMA 2 bytes Array: nscan2
- HTR_STAT_RDA 2 bytes Array: nscan2
- HTR_STAT_RS 2 bytes Array: nscan2
- PWR_STAT_LR_RED 2 bytes Array: nscan2
- RS_MST_RLY_STAT 2 bytes Array: nscan2
- RDA_DEPL_STAT_4 2 bytes Array: nscan2
- RDA_DEPL_STAT_3 2 bytes Array: nscan2
- RDA_DEPL_STAT_2 2 bytes Array: nscan2
- RDA_DEPL_STAT_1 2 bytes Array: nscan2
- CLR_STAT_N 2 bytes Array: nscan2
- IBS_LR3_STAT_N 2 bytes Array: nscan2
- MR_LR_LOWSTAT 2 bytes Array: nscan2
- IBS_LR2_STAT_N 2 bytes Array: nscan2
- MR_LR_LEFT_STAT 2 bytes Array: nscan2
- IBS_LR1_STAT_N 2 bytes Array: nscan2
- MR_LR_RIGHT_STAT 2 bytes Array: nscan2
- IE_LATCHUP_CHAN 2 bytes Array: nscan2
- IE_LATCHUP_RETR 2 bytes Array: nscan2
- IE_LATCHUP_SMPS 2 bytes Array: nscan2
- IE_LATCHUP_NUM 2 bytes Array: nscan2

continued on next figure

•

•

•
continued from last figure

Figure 41: Data Format Structure for 1AGMI, S3, IE_TELEMETRY
5.1 1AGMI - GMI unpacked packet data

![Diagram showing data format structure for 1AGMI, MECHANISMS with labels for various fields such as SCE_LAST_CMD, SCE_LAST_RESPNS, SCE_CMD_CNT, SMA_RTPRB_SEC, SMA_RATE, SCE_RATE, SCE_CMD_RATE, SMA_CMD_RATE, RESOLVER_POS, MECH_CMD_CNT, SCE_INHIBIT, TACH_PULSE_CNT, LR_ABRT_CNT, RAMP_ABRT_CNT, OVRD_RDA_LR, OVRD_IBS_LR, SCE_PASSPROT, RAMP_INPROGRESS, SMA_SPINNING, LR_RLS_IN_PROG, SCE_A_POWER, SCE_B_POWER, SCE_SELECTION, SMA_RATE_PROB, and an array 'nscan2'.]

continued on next figure

Figure 42: Data Format Structure for 1AGMI, MECHANISMS
Figure 43: Data Format Structure for 1AGMI, S3, MECHANISMS
5.1 1AGMI - GMI unpacked packet data

Figure 44: Data Format Structure for 1AGMI, SMPL_INFO

continued on next figure
Figure 45: Data Format Structure for 1AGMI, S3, SMPL_INFO
Figure 46: Data Format Structure for 1AGMI, S4, ScanTime
**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**gmi1aHeader** (Group)

**sampleRangeFile** (2-byte unsigned integer, array size: dim6 x dim7):
The sample range table that was used to subset S1 and S2.
5.1 1AGMI - GMI unpacked packet data

S1 (Swath)

S1_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S1)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
  -9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
  -99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
  -99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
  -99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
  -99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
  -99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
  -9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
  -9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
  -99999.9 Missing value

Latitude (4-byte float, array size: npixelev x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixelev x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S1)

**dataQuality** (1-byte integer, array size: nsan1):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$, the unsigned integer value is $2^i$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit $= 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nsan1):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit $= 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nsan1):
A summary of status modes. If all status modes are routine, all bits in modeStatus $= 0$. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit $= 0$ if the status is routine but the bit $= 1$ if the status is not routine. Bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$, the unsigned integer value is $2^i$). The non-routine situations follow:
5.1 1AGMI - GMI unpacked packet data

Bit Meaning if bit = 1
0  Spare (always 0)
1  SCorientation not 0 or 180
2  pointingStatus not 0
3  Spare (always 0)
4  Non-routine operationalMode
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

geoError (2-byte integer, array size: nscan1):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0  Latitude limit exceeded for viewed pixel locations
1  Negative scan time, invalid input
2  Error getting spacecraft attitude at scan mid-time
3  Error getting spacecraft ephemeris at scan mid-time
4  Invalid input non-unit ray vector for any pixel
5  Ray misses Earth for any pixel with normal pointing
6  Nadir calculation error for subsatellite position
7  Pixel count with geolocation error over threshold
8  Error in getting spacecraft attitude for any pixel
9  Error in getting spacecraft ephemeris for any pixel
10  Spare (always 0)
11  Spare (always 0)
12  Spare (always 0)
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

geoWarning (2-byte integer, array size: nscan1):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be
useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit \(i = 1\) and other bits = 0 the unsigned integer value is \(2^i\)):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan1):
The positive angle of the spacecraft vector \(v\) from the satellite forward direction of motion, measured clockwise facing down. We define \(v\) in the same direction as the spacecraft axis \(+X\), which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(+X) forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>(-X) forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan1):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
acsModeMidScan (1-byte integer, array size: nscan1):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan1):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan1):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

ephemerisUsed (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
255  Missing value
navigation (Group in S1)

scPos (4-byte float, array size: XYZ x nscan1):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan1):
The velocity vector $(ms^{-1})$ of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan1):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan1):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will
show an orbital frequency component relative to the Earth fixed ground track due to the 
Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. 
Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan1):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan1):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value
**timeMidScanOffset** (8-byte float, array size: nscan1):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**sunData** (Group in S1)

**solarBetaAngle** (4-byte float, array size: nscan1):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

**phaseFromOrbitMidnight** (4-byte float, array size: nscan1):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

**sunEarthSeparation** (4-byte float, array size: nscan1):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

**earthAngularRadius** (4-byte float, array size: nscan1):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**phaseOfEclipseExit** (4-byte float, array size: nscan1):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**orbitRate** (4-byte float, array size: nscan1):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan1):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values
range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan1):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npixelev x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npixelev x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: npixelev x nscan1):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: npixelev x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npixelev x nscan1):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**moonVectorInstFrame** (4-byte float, array size: GMIxyz x nscan1):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
-9999.9 Missing value

**earthView** (2-byte unsigned integer, array size: nchannel1 x npixelev x nscan1):
Earth view counts.

Special values are defined as:
  0  Missing value

**hotLoad** (2-byte unsigned integer, array size: nchannel1 x npixelht x nscan1):

Hot load counts.

Special values are defined as:
  0  Missing value

**coldSky** (2-byte unsigned integer, array size: nchannel1 x npixelcs x nscan1):

Cold sky counts.

Special values are defined as:
  0  Missing value

**earthViewBlanking** (1-byte char, array size: VH x npixelev x nscan1):

Earth view blanking counts.

Special values are defined as:
  0  Missing value

**hotLoadBlanking** (1-byte char, array size: VH x npixelht x nscan1):

Hot load blanking counts.

Special values are defined as:
  0  Missing value

**coldSkyBlanking** (1-byte char, array size: VH x npixelcs x nscan1):

Cold sky blanking counts.
5.1 1AGMI - GMI unpacked packet data

Special values are defined as:
0 Missing value

S2 (Swath)

S2_S swathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2
data products. See Metadata for GPM Products for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-99999 Missing value
**SecondOfDay** (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npixelev x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixelev x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S2)

**dataQuality** (1-byte integer, array size: nscan2):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan2):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
5.1 1AGMI - GMI unpacked packet data

**modeStatus** (1-byte integer, array size: nscan2):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorient not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan2):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan2):
A summary of geolocation warnings in the scan. **geoWarning** does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCOrientation** (2-byte integer, array size: nscan2):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan2):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is
good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

Value Meaning
0 Nominal pointing in Mission Science Mode
1 GPS point solution stale and PVT ephemeris used
2 GEONS solution stale and GEONS ephemeris used
-8000 Non-nominal mission science orientation
-9999 Missing

acsModeMidScan (1-byte integer, array size: nscan2):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

Value Meaning
0 LAUNCH
1 RATENULL
2 SUNPOINT
3 GSPM (Gyro-less Sun Point)
4 MSM (Mission Science Mode)
5 SLEW
6 DELTAH
7 DELTAV
-99 UNKNOWN -- ACS mode unavailable

targetSelectionMidScan (1-byte integer, array size: nscan2):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

Value Meaning
0 S/C Z axis nadir, +X in flight direction
1 Flight Z axis nadir, +X in flight direction
2 S/C Z axis nadir, -X in flight direction
3 Flight Z axis nadir, -X in flight direction
4 +90 yaw for DPR antenna pattern calibration
5 -90 yaw for DPR antenna pattern calibration
-99 Missing

operationalMode (1-byte integer, array size: nscan2):
Status of the GMI instrument.

Bit Meaning if bit = 1
0 Receiver status (0=ON, 1=OFF)
1 Spinup Status (0=ON, 1=OFF)
**FractionalGranuleNumber** (8-byte float, array size: nscan2):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**ephemerisUsed** (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
255 Missing value

**navigation** (Group in S2)

**scPos** (4-byte float, array size: XYZ x nscan2):
The position vector(m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordi-
nates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan
period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

**scVel** (4-byte float, array size: XYZ x nscan2):
The velocity vector \((\text{m s}^{-1})\) of the spacecraft in ECEF Coordinates at the Scan mid-Time.
Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

**scLat** (4-byte float, array size: nscan2):
The geodeic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values
range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

**scLon** (4-byte float, array size: nscan2):
The geodeic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values
range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAlt** (4-byte float, array size: nscan2):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values
range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan2):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from
DPR science telemetry. This is empty in non-DPR products. Values range from 350000
to 500000 m. Special values are defined as:
-9999.9 Missing value
scAttRollGeoc (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeoc (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeoc (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeod (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeod (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value
greenHourAng (4-byte float, array size: nscan2):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan2):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan2):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

sunData (Group in S2)

solarBetaAngle (4-byte float, array size: nscan2):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan2):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan2):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan2):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan2):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow,
based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**orbitRate** (4-byte float, array size: nscan2):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan2):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan2):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npixelev x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npixelev x nscan2):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: npixelev x nscan2):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: npixelev x nscan2):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npixelev x nscan2):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsiod-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like)
sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**moonVectorInstFrame** (4-byte float, array size: GMIxyz x nscan2):
The x, y, z components of the moon vector in the GMI instrument coordinate system.
Values are in counts. Special values are defined as:
-9999.9 Missing value

**earthView** (2-byte unsigned integer, array size: nchannel2 x npixelev x nscan2):
Earth view counts.

Special values are defined as:
0  Missing value

**hotLoad** (2-byte unsigned integer, array size: nchannel2 x npixelht x nscan2):
Hot load counts.

Special values are defined as:
0  Missing value

**coldSky** (2-byte unsigned integer, array size: nchannel2 x npixelcs x nscan2):
Cold sky counts.

Special values are defined as:
0  Missing value

**S3** (Swath)

**S3_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S3)
A UTC time associated with the scan.
5.1 1AGMI - GMI unpacked packet data

**Year** (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**ephemerisUsed** (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
255 Missing value

**Latitude** (4-byte float, array size: nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**TAM1** (Group in S3)

- **timeOffset** (4-byte float, array size: nscan2):
  Time offset between magnetometer and scan time. Values range from -100 to 100 s. Special values are defined as:
  -9999.9 Missing value

- **Vx** (2-byte unsigned integer, array size: nscan2):
  Magnetometer one vector, x component. Values range from 0 to 65535 count. Special values are defined as:
  65535 Missing value

- **Vy** (2-byte unsigned integer, array size: nscan2):
  Magnetometer one vector, y component. Values range from 0 to 65535 count. Special values are defined as:
  65535 Missing value

- **Vz** (2-byte unsigned integer, array size: nscan2):
  Magnetometer one vector, z component. Values range from 0 to 65535 count. Special values are defined as:
  65535 Missing value

**TAM2** (Group in S3)

- **timeOffset** (4-byte float, array size: nscan2):
  Time offset between magnetometer and scan time. Values range from -100 to 100 s. Special values are defined as:
  -9999.9 Missing value

- **Vx** (2-byte unsigned integer, array size: nscan2):
  Magnetometer two vector, x component. Values range from 0 to 65535 count. Special values are defined as:
  65535 Missing value

- **Vy** (2-byte unsigned integer, array size: nscan2):
  Magnetometer two vector, y component. Values range from 0 to 65535 count. Special values are defined as:
  65535 Missing value

- **Vz** (2-byte unsigned integer, array size: nscan2):
  Magnetometer two vector, z component. Values range from 0 to 65535 count. Special
values are defined as:
65535 Missing value

**TORQUE_BAR** (Group in S3)

`timeOffset` (4-byte float, array size: nscan2):
Time offset between torque bar and scan time. Values range from -100 to 100 s. Special values are defined as:
-9999.9 Missing value

`Vx` (2-byte unsigned integer, array size: nscan2):
Torque bar vector, x component. Values range from 0 to 65535 count. Special values are defined as:
65535 Missing value

`Vy` (2-byte unsigned integer, array size: nscan2):
Torque bar vector, y component. Values range from 0 to 65535 count. Special values are defined as:
65535 Missing value

`Vz` (2-byte unsigned integer, array size: nscan2):
Torque bar vector, z component. Values range from 0 to 65535 count. Special values are defined as:
65535 Missing value

**GMI_TEMPERATURES** (Group in S3)

`timeOffset` (4-byte float, array size: nscan2):
Time offset between thermisters and scan time. Values range from -100 to 100 s. Special values are defined as:
-9999.9 Missing value

`apid` (2-byte unsigned integer, array size: nscan2):
APID. 0 is the missing value.

`SMA_PT_TEMP` (2-byte unsigned integer, array size: nscan2):
SMA_PT_TEMP. 0 is the missing value.

`ICA_PT_TEMP` (2-byte unsigned integer, array size: nscan2):
ICA_PT_TEMP. 0 is the missing value.

`RS_PT_TEMP` (2-byte unsigned integer, array size: nscan2):
RS_PT_TEMP. 0 is the missing value.
STAT_PT_TEMP (2-byte unsigned integer, array size: nscan2):
STAT_PT_TEMP. 0 is the missing value.

MR_PT_TEMP (2-byte unsigned integer, array size: nscan2):
MR_PT_TEMP. 0 is the missing value.

primaryHeader (Group in S3)

version (1-byte integer, array size: nscan2):

type (1-byte integer, array size: nscan2):

secHeaderFlag (1-byte integer, array size: nscan2):

APID (2-byte integer, array size: nscan2):

sequenceFlag (1-byte integer, array size: nscan2):

packetSequenceCount (2-byte integer, array size: nscan2):

packetLength (2-byte unsigned integer, array size: nscan2):

instrTimeSeconds (4-byte unsigned integer, array size: nscan2):

instrTimeSubseconds (2-byte unsigned integer, array size: nscan2):

numPacketSegments (1-byte integer, array size: nscan2):

spare (1-byte integer, array size: nscan2):

RDRversion (2-byte integer, array size: nscan2):

GSDR_TIME (Group in S3)
\textbf{G\_TC\_PULSE\_SECS} (4-byte unsigned integer, array size: nscan2):
GMI Instrument Time Code Pulse Timestamp (Seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{G\_TC\_PULSE\_SUBS} (2-byte unsigned integer, array size: nscan2):
GMI Instrument Time Code Pulse Timestamp (Sub-Seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{G\_TCU\_SECS} (4-byte unsigned integer, array size: nscan2):
S/C Time Code Update (Seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{G\_TCU\_SUBS} (4-byte unsigned integer, array size: nscan2):
S/C Time Code Update (Sub-Seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{G\_TCF\_SC\_SECS} (4-byte unsigned integer, array size: nscan2):
Time Correlation Factor spacecraft timestamp (Seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{G\_TCF\_SC\_SUBSEC} (4-byte unsigned integer, array size: nscan2):
Time Correlation Factor spacecraft timestamp (Sub-seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{G\_TCF\_SECS} (4-byte unsigned integer, array size: nscan2):
Time Correlation Factor (Seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{G\_TCF\_SUBSECS} (4-byte unsigned integer, array size: nscan2):
Time Correlation Factor (Sub-seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{G\_TCF\_SIGN} (2-byte unsigned integer, array size: nscan2):
Time Correlation Factor (Sign) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{G\_TCF\_LEAP} (2-byte unsigned integer, array size: nscan2):
Time Correlation Factor (Leap Seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{GPS\_TCU\_SECS} (4-byte unsigned integer, array size: nscan2):
S/C Time Code Update in GPS time (Seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}

\textbf{GPS\_TCU\_SUBS} (4-byte unsigned integer, array size: nscan2):
S/C Time Code Update in GPS time (Sub-Seconds) */ Special values are defined as:
\begin{itemize}
\item 0 Missing value
\end{itemize}
SENSOR_INFO (Group in S3)

**KEEP_ALIVE_CNT** (4-byte unsigned integer, array size: nscan2):
Keep Alive Counter */ Special values are defined as:
  0 Missing value

**FPGA_RST_REASON** (2-byte unsigned integer, array size: nscan2):
Reason for last reset */ Special values are defined as:
  0 Missing value

**CRASH_REASON** (2-byte unsigned integer, array size: nscan2):
Reason for last crash */ Special values are defined as:
  0 Missing value

**VERSION_MIN** (2-byte unsigned integer, array size: nscan2):
GMI FSW minor version number */ Special values are defined as:
  0 Missing value

**VERSION_MAG** (2-byte unsigned integer, array size: nscan2):
GMI FSW major version number */ Special values are defined as:
  0 Missing value

**FPGA_MODE** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0 Missing value

**ERR_hdl_FAILURE** (2-byte unsigned integer, array size: nscan2):
Error Handler Failure Flag */ Special values are defined as:
  0 Missing value

**RESET_REASON** (2-byte unsigned integer, array size: nscan2):
Reason for last reset */ Special values are defined as:
  0 Missing value

**BOOT_BANK** (2-byte unsigned integer, array size: nscan2):
EEPROM Bank of last reboot */ Special values are defined as:
  0 Missing value

**CURRENT_BANK** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0 Missing value

**EDAC_ENABLE** (2-byte unsigned integer, array size: nscan2):
EDAC enable */ Special values are defined as:
  0 Missing value

**WDOG_ENABLE** (2-byte unsigned integer, array size: nscan2):
FPGA CNTL Special values are defined as:
  0 Missing value
SC_1HZ_REF (2-byte unsigned integer, array size: nscan2):
FPGA CNTRL Special values are defined as:
  0 Missing value

SCE_FORCE_SEL (2-byte unsigned integer, array size: nscan2):
FPGA CNTRL Special values are defined as:
  0 Missing value

RS_1MHZ_REF (2-byte unsigned integer, array size: nscan2):
FPGA CNTRL Special values are defined as:
  0 Missing value

RS_SCAN_START (2-byte unsigned integer, array size: nscan2):
FPGA CNTRL Special values are defined as:
  0 Missing value

FPGA_IE_RX_EN (2-byte unsigned integer, array size: nscan2):
FPGA CNTRL Special values are defined as:
  0 Missing value

FPGA_RS_RX_EN (2-byte unsigned integer, array size: nscan2):
FPGA CNTRL Special values are defined as:
  0 Missing value

FPGA_SCE_RX_EN (2-byte unsigned integer, array size: nscan2):
FPGA CNTRL Special values are defined as:
  0 Missing value

EEPROM_BUSY (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0 Missing value

RS_TLM_PROG (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0 Missing value

SCE_A_ACTIVE (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0 Missing value

SCE_A_RLY (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0 Missing value

SCE_B_ACTIVE (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0 Missing value

SCE_B_RLY (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0 Missing value
**IE_TLM_PROG** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0  Missing value

**SCE_RSP_PROG** (2-byte unsigned integer, array size: nscan2):
SCE response in progress. */ Special values are defined as:
  0  Missing value

**RS_CLK_ERR** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0  Missing value

**RS_PKT_ERR** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0  Missing value

**RS_TLM_ERR** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0  Missing value

**SCE_RSP_RDY** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0  Missing value

**IE_PKT_ERR** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0  Missing value

**IE_CMD_ERR** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0  Missing value

**IE_RSP_ERR** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0  Missing value

**IE_TLM_ERR** (2-byte unsigned integer, array size: nscan2):
IE Tlm Error FPGA status bit */ Special values are defined as:
  0  Missing value

**FPGA_ACCESS_ERR** (2-byte unsigned integer, array size: nscan2):
FPGA STAT Special values are defined as:
  0  Missing value

**RS_INFO** (Group in S3)

**RS_POWERED** (2-byte unsigned integer, array size: nscan2):
RS Power */ Special values are defined as:
  0  Missing value
5.1 AGMI - GMI unpacked packet data

**RS_ENABLED** (2-byte unsigned integer, array size: nscan2):
RS Science Enabled */ Special values are defined as:
  0 Missing value

**RS_MST_RLY** (2-byte unsigned integer, array size: nscan2):
RS Master Relay */ Special values are defined as:
  0 Missing value

**RS_10GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
RS 10GHz Relay */ Special values are defined as:
  0 Missing value

**RS_18GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
RS 18GHz Relay */ Special values are defined as:
  0 Missing value

**RS_23GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
RS 23GHz Relay */ Special values are defined as:
  0 Missing value

**RS_36GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
RS 36GHz Relay */ Special values are defined as:
  0 Missing value

**RS_89GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
RS 89GHz Relay */ Special values are defined as:
  0 Missing value

**RS_166GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
RS 166GHz Relay */ Special values are defined as:
  0 Missing value

**RS_183GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
RS 183GHz Relay */ Special values are defined as:
  0 Missing value

**RS_DQ_MISSING** (2-byte unsigned integer, array size: nscan2):
RS Data Quality indicator (Missing samples) */ Special values are defined as:
  0 Missing value

**RS_DQ_EXTRAS** (2-byte unsigned integer, array size: nscan2):
RS Data Quality indicator (Extra samples) */ Special values are defined as:
  0 Missing value

**RS_DQ_DUPES** (2-byte unsigned integer, array size: nscan2):
RS Data Quality indicator (Duplicate samples) */ Special values are defined as:
  0 Missing value

**RS_LAST_REV** (2-byte unsigned integer, array size: nscan2):
RS Data Quality indicator (Latest Revolution) */ Special values are defined as:
  0 Missing value
**RS_DQSAME_REV** (2-byte unsigned integer, array size: nscan2): RS Data Quality indicator (Revolution bit not changing) */ Special values are defined as:

0 Missing value

**RS_DQBAD_REVS** (2-byte unsigned integer, array size: nscan2): RS Data Quality indicator (Inconsistent Revolutions) */ Special values are defined as:

0 Missing value

**RS_DQPAR_ERR** (2-byte unsigned integer, array size: nscan2): RS Data Quality indicator (Parity Error) */ Special values are defined as:

0 Missing value

**RS_DQCLK_ERR** (2-byte unsigned integer, array size: nscan2): RS Data Quality indicator (Clock Error) */ Special values are defined as:

0 Missing value

**RS_DQPKT_ERR** (2-byte unsigned integer, array size: nscan2): RS Data Quality indicator (Packet Error) */ Special values are defined as:

0 Missing value

**RS_DQTLM_ERR** (2-byte unsigned integer, array size: nscan2): RS Data Quality indicator (Telemetry Error) */ Special values are defined as:

0 Missing value

**RS_DQBAD_CONF** (2-byte unsigned integer, array size: nscan2): RS Data Quality indicator (Mismatched configuration) */ Special values are defined as:

0 Missing value

**RS_DQCAL_LIM** (2-byte unsigned integer, array size: nscan2): RS Data Quality indicator (Calibration Limits) */ Special values are defined as:

0 Missing value

**BLK_STATE** (2-byte unsigned integer, array size: nscan2): Blanking State */ Special values are defined as:

0 Missing value

**BLK_SIDE** (2-byte unsigned integer, array size: nscan2): Blanking Side */ Special values are defined as:

0 Missing value

**BLK_DELAY** (2-byte unsigned integer, array size: nscan2): Blanking Delay */ Special values are defined as:

0 Missing value

**BLK_DURATION** (2-byte unsigned integer, array size: nscan2): Blanking Duration */ Special values are defined as:

0 Missing value

**RS_HSK_SIZE** (2-byte unsigned integer, array size: nscan2): The number of RS Housekeeping samples */ Special values are defined as:

0 Missing value
RS_PAR_ERR_CNT (4-byte unsigned integer, array size: nscan2): Number of RS parity errors */ Special values are defined as:
0   Missing value

RS_SCAN_CNT (4-byte unsigned integer, array size: nscan2): Number of RS scans */ Special values are defined as:
0   Missing value

SAMPLE_TBL_VER (4-byte unsigned integer, array size: nscan2): Sample table version */ Special values are defined as:
0   Missing value

SMPL_TBL (4-byte unsigned integer, array size: nscan2): Sample Table Pointer */ Special values are defined as:
0   Missing value

RS_SC_SIZE (2-byte unsigned integer, array size: nscan2): The number of science samples */ Special values are defined as:
0   Missing value

GSDR_SIZE (2-byte unsigned integer, array size: nscan2): The size of the latest GSDR */ Special values are defined as:
0   Missing value

GSDR_LEFT (2-byte unsigned integer, array size: nscan2): GSDR Remainder */ Special values are defined as:
0   Missing value

GSDR_B_POP_ID (2-byte unsigned integer, array size: nscan2): GSDR Buffer Pool Index (Pop) */ Special values are defined as:
0   Missing value

GSDR_B_PUSH_ID (2-byte unsigned integer, array size: nscan2): GSDR Buffer Pool Index (Push) */ Special values are defined as:
0   Missing value

GSDR_API (2-byte unsigned integer, array size: nscan2): GSDR Apid */ Special values are defined as:
0   Missing value

PKT_STATE (2-byte unsigned integer, array size: nscan2): Packetizing State */ Special values are defined as:
0   Missing value

OVRD_RS_PWR (2-byte unsigned integer, array size: nscan2): Override RS Power Check */ Special values are defined as:
0   Missing value

OVRD_SMA_SPIN (2-byte unsigned integer, array size: nscan2): Override SMA Spinning Check */ Special values are defined as:
0   Missing value
**OVRD_PASS_RS** (2-byte unsigned integer, array size: nscan2):
Override RS Passthru protection indicator. */ Special values are defined as:
0  Missing value

**SYNCH_STAMPS** (Group in S3)

**IDX_PULSE_SECS** (4-byte unsigned integer, array size: nscan2):
Index Pulse (Seconds) */ Special values are defined as:
0  Missing value

**IDX_PULSE_SUBS** (2-byte unsigned integer, array size: nscan2):
Index Pulse (Sub-Seconds) */ Special values are defined as:
0  Missing value

**TACH_SECS_00** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 0 */ Special values are defined as:
0  Missing value

**TACH_SUBS_00** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 0 */ Special values are defined as:
0  Missing value

**TACH_SECS_01** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 1 */ Special values are defined as:
0  Missing value

**TACH_SUBS_01** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 1 */ Special values are defined as:
0  Missing value

**TACH_SECS_02** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 2 */ Special values are defined as:
0  Missing value

**TACH_SUBS_02** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 2 */ Special values are defined as:
0  Missing value

**TACH_SECS_03** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 3 */ Special values are defined as:
0  Missing value

**TACH_SUBS_03** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 3 */ Special values are defined as:
0  Missing value

**TACH_SECS_04** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 4 */ Special values are defined as:
0  Missing value
5.1 IAGMI - GMI unpacked packet data

TACH_SUBS_04 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 4 */ Special values are defined as:
0 Missing value

TACH_SECS_05 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 5 */ Special values are defined as:
0 Missing value

TACH_SUBS_05 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 5 */ Special values are defined as:
0 Missing value

TACH_SECS_06 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 6 */ Special values are defined as:
0 Missing value

TACH_SUBS_06 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 6 */ Special values are defined as:
0 Missing value

TACH_SECS_07 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 7 */ Special values are defined as:
0 Missing value

TACH_SUBS_07 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 7 */ Special values are defined as:
0 Missing value

TACH_SECS_08 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 8 */ Special values are defined as:
0 Missing value

TACH_SUBS_08 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 8 */ Special values are defined as:
0 Missing value

TACH_SECS_09 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 9 */ Special values are defined as:
0 Missing value

TACH_SUBS_09 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 9 */ Special values are defined as:
0 Missing value

TACH_SECS_10 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 10 */ Special values are defined as:
0 Missing value

TACH_SUBS_10 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 10 */ Special values are defined as:
0 Missing value
TACH_SECS_11 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 11 */ Special values are defined as:
  0  Missing value
TACH_SUBS_11 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 11 */ Special values are defined as:
  0  Missing value
TACH_SECS_12 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 12 */ Special values are defined as:
  0  Missing value
TACH_SUBS_12 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 12 */ Special values are defined as:
  0  Missing value
TACH_SECS_13 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 13 */ Special values are defined as:
  0  Missing value
TACH_SUBS_13 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 13 */ Special values are defined as:
  0  Missing value
TACH_SECS_14 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 14 */ Special values are defined as:
  0  Missing value
TACH_SUBS_14 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 14 */ Special values are defined as:
  0  Missing value
TACH_SECS_15 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 15 */ Special values are defined as:
  0  Missing value
TACH_SUBS_15 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 15 */ Special values are defined as:
  0  Missing value

SYNCH_STAMPS2 (Group in S3)

TACH_SECS_16 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 16 */ Special values are defined as:
  0  Missing value
TACH_SUBS_16 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 16 */ Special values are defined as:
  0  Missing value
TACH_SECS_17 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 17 */ Special values are defined as:
  0  Missing value

TACH_SUBS_17 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 17 */ Special values are defined as:
  0  Missing value

TACH_SECS_18 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 18 */ Special values are defined as:
  0  Missing value

TACH_SUBS_18 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 18 */ Special values are defined as:
  0  Missing value

TACH_SECS_19 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 19 */ Special values are defined as:
  0  Missing value

TACH_SUBS_19 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 19 */ Special values are defined as:
  0  Missing value

TACH_SECS_20 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 20 */ Special values are defined as:
  0  Missing value

TACH_SUBS_20 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 20 */ Special values are defined as:
  0  Missing value

TACH_SECS_21 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 21 */ Special values are defined as:
  0  Missing value

TACH_SUBS_21 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 21 */ Special values are defined as:
  0  Missing value

TACH_SECS_22 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 22 */ Special values are defined as:
  0  Missing value

TACH_SUBS_22 (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 22 */ Special values are defined as:
  0  Missing value

TACH_SECS_23 (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 23 */ Special values are defined as:
  0  Missing value
**TACH_SUBS.23** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 23 */ Special values are defined as:
  0 Missing value

**TACH_SECS.24** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 24 */ Special values are defined as:
  0 Missing value

**TACH_SUBS.24** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 24 */ Special values are defined as:
  0 Missing value

**TACH_SECS.25** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 25 */ Special values are defined as:
  0 Missing value

**TACH_SUBS.25** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 25 */ Special values are defined as:
  0 Missing value

**TACH_SECS.26** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 26 */ Special values are defined as:
  0 Missing value

**TACH_SUBS.26** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 26 */ Special values are defined as:
  0 Missing value

**TACH_SECS.27** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 27 */ Special values are defined as:
  0 Missing value

**TACH_SUBS.27** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 27 */ Special values are defined as:
  0 Missing value

**TACH_SECS.28** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 28 */ Special values are defined as:
  0 Missing value

**TACH_SUBS.28** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 28 */ Special values are defined as:
  0 Missing value

**TACH_SECS.29** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 29 */ Special values are defined as:
  0 Missing value

**TACH_SUBS.29** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 29 */ Special values are defined as:
  0 Missing value
5.1 1AGMI - GMI unpacked packet data

**TACH_SECS_30** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 30 */ Special values are defined as:
0  Missing value

**TACH_SUBS_30** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 30 */ Special values are defined as:
0  Missing value

**TACH_SECS_31** (4-byte unsigned integer, array size: nscan2):
Tachometer Pulse Seconds 31 */ Special values are defined as:
0  Missing value

**TACH_SUBS_31** (2-byte unsigned integer, array size: nscan2):
Tachometer Pulse Subseconds 31 */ Special values are defined as:
0  Missing value

**SCAN_COMPL_SECS** (4-byte unsigned integer, array size: nscan2):
Scan Complete Time Tag Seconds */ Special values are defined as:
0  Missing value

**SCAN_COMPL_SUBS** (2-byte unsigned integer, array size: nscan2):
Scan Complete Time Tag Subseconds */ Special values are defined as:
0  Missing value

**RSHSK_STATUS** (Group in S3)

**RSST_SCI_ADC_LP** (2-byte unsigned integer, array size: nscan2):
RS Science channel latchup */ Special values are defined as:
0  Missing value

**RSST_HSK_ADC_LP** (2-byte unsigned integer, array size: nscan2):
RS Housekeeping channel ADC latchup */ Special values are defined as:
0  Missing value

**RSST_SAMP_OVLP** (2-byte unsigned integer, array size: nscan2):
Sample offset overlap */ Special values are defined as:
0  Missing value

**RSST_10GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
10 GHz relay status */ Special values are defined as:
0  Missing value

**RSST_18GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
18 GHz relay status */ Special values are defined as:
0  Missing value

**RSST_23GHZ_RLY** (2-byte unsigned integer, array size: nscan2):
23 GHz relay status */ Special values are defined as:
0  Missing value
RSST_36GHZ_RLY (2-byte unsigned integer, array size: nscan2):
36 GHz relay status */ Special values are defined as:
  0  Missing value
RSST_89GHZ_RLY (2-byte unsigned integer, array size: nscan2):
89 GHz relay status */ Special values are defined as:
  0  Missing value
RSST_166GHZ_RLY (2-byte unsigned integer, array size: nscan2):
166 GHz relay status */ Special values are defined as:
  0  Missing value
RSST_183GHZ_RLY (2-byte unsigned integer, array size: nscan2):
183 GHz relay status */ Special values are defined as:
  0  Missing value
RSST_INVLD_CMD (2-byte unsigned integer, array size: nscan2):
Invalid command received */ Special values are defined as:
  0  Missing value
RSST_CMD_AFTER (2-byte unsigned integer, array size: nscan2):
Command received after scan start */ Special values are defined as:
  0  Missing value
NDIODE_MODE (2-byte unsigned integer, array size: nscan2):
RS Configuration of Noise Diode Mode */ Special values are defined as:
  0  Missing value
RSST_NDIODE_ST (2-byte unsigned integer, array size: nscan2):
Noise diode state during the scan */ Special values are defined as:
  0  Missing value
NDIODE10GHZSNUM (2-byte unsigned integer, array size: nscan2):
RS Configuration of Noise Diode Start Sample Number */ Special values are defined as:
  0  Missing value
RESERVED1 (2-byte unsigned integer, array size: nscan2):
Unused item */ Special values are defined as:
  0  Missing value
RS_CALRES_1 (2-byte unsigned integer, array size: nscan2):
RS Calibration Resistor for RS telemetry num 1 */ Special values are defined as:
  0  Missing value
BATC_CALRES_1 (2-byte unsigned integer, array size: nscan2):
RS Calibration Resistor for BATC telemetry num 1 */ Special values are defined as:
  0  Missing value
RS_CALRES_2 (2-byte unsigned integer, array size: nscan2):
RS Calibration Resistor for BATC telemetry num 2 */ Special values are defined as:
  0  Missing value
5.1 IAGMI - GMI unpacked packet data

**BATC_CALRES_2** (2-byte unsigned integer, array size: nscan2):
RS Calibration Resistor for BATC telemetry num 2 */ Special values are defined as:
0 Missing value

**RS_EPC_ISENS** (2-byte unsigned integer, array size: nscan2):
Receiver Subsystem EPC Current Sense */ Special values are defined as:
0 Missing value

**RS_EPC_5V** (2-byte unsigned integer, array size: nscan2):
EPC 5V Telemetry */ Special values are defined as:
0 Missing value

**RS_EPC_7V** (2-byte unsigned integer, array size: nscan2):
EPC 7V Telemetry */ Special values are defined as:
0 Missing value

**RS_EPC_POS12V** (2-byte unsigned integer, array size: nscan2):
EPC +12V Telemetry */ Special values are defined as:
0 Missing value

**RS_EPC_NEG12V** (2-byte unsigned integer, array size: nscan2):
EPC -12V Telemetry */ Special values are defined as:
0 Missing value

**RS_EPC_15V** (2-byte unsigned integer, array size: nscan2):
EPC 15V Telemetry */ Special values are defined as:
0 Missing value

**RSHSK_SAMPL_INFO** (Group in S3)

**SMPOFFST_10GHZ** (2-byte unsigned integer, array size: nscan2):
RS-Reported Sample Offset for the 10GHz Channels (4us) */ Special values are defined as:
0 Missing value

**SMPOFFST_18GHZ** (2-byte unsigned integer, array size: nscan2):
RS-Reported Sample Offset for the 18 GHz Channels (4us) */ Special values are defined as:
0 Missing value

**SMPOFFST_23GHZ** (2-byte unsigned integer, array size: nscan2):
RS-Reported Sample Offset for the 23 GHz Channel (4us) */ Special values are defined as:
0 Missing value

**SMPOFFST_36GHZ** (2-byte unsigned integer, array size: nscan2):
RS-Reported Sample Offset for the 36 GHz Channels (4us) */ Special values are defined
as:
  0  Missing value

**SMPOFFST_89GHZ** (2-byte unsigned integer, array size: nscan2): RS-Reported Sample Offset for the 89 GHz Channels (4us) */ Special values are defined as:
  0  Missing value

**SMPOFFST_166GHZ** (2-byte unsigned integer, array size: nscan2): RS-Reported Sample Offset for the 166 GHz Channels (4us) */ Special values are defined as:
  0  Missing value

**SMPOFFST_183GHZ** (2-byte unsigned integer, array size: nscan2): RS-Reported Sample Offset for the 183 GHz Channels (4us) */ Special values are defined as:
  0  Missing value

**NUMSMPLS_10GHZ** (2-byte unsigned integer, array size: nscan2): RS Configuration of Number of Samples for the 10 GHz channels */ Special values are defined as:
  0  Missing value

**NUMSMPLS_18GHZ** (2-byte unsigned integer, array size: nscan2): RS Configuration of Number of Samples for the 10 GHz channels */ Special values are defined as:
  0  Missing value

**NUMSMPLS_23GHZ** (2-byte unsigned integer, array size: nscan2): RS Configuration of Number of Samples for the 10 GHz channels */ Special values are defined as:
  0  Missing value

**NUMSMPLS_36GHZ** (2-byte unsigned integer, array size: nscan2): RS Configuration of Number of Samples for the 10 GHz channels */ Special values are defined as:
  0  Missing value

**NUMSMPLS_89GHZ** (2-byte unsigned integer, array size: nscan2): RS Configuration of Number of Samples for the 10 GHz channels */ Special values are defined as:
  0  Missing value

**NUMSMPLS_166GHZ** (2-byte unsigned integer, array size: nscan2): RS Configuration of Number of Samples for the 166 GHz channels */ Special values are defined as:
  0  Missing value

**NUMSMPLS_183GHZ** (2-byte unsigned integer, array size: nscan2): RS Configuration of Number of Samples for the 183 GHz channels */ Special values are
5.1 1AGMI - GMI unpacked packet data

defined as:
  0  Missing value

RSHSK_GAIN (Group in S3)

RESERVED2 (2-byte unsigned integer, array size: nscan2):
Unused item */ Special values are defined as:
  0  Missing value

GAIN_V10GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 10 GHz V-pol channel */ Special values are defined as:
  0  Missing value

GAIN_H36GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 36 GHz H-pol channel */ Special values are defined as:
  0  Missing value

GAIN_H89GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 89 GHz H-pol channel */ Special values are defined as:
  0  Missing value

GAIN_H10GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 10 GHz H-pol channel */ Special values are defined as:
  0  Missing value

GAIN_H166GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 166 GHz H-pol channel */ Special values are defined as:
  0  Missing value

GAIN_V18GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 18 GHz V-pol channel */ Special values are defined as:
  0  Missing value

GAIN_H18GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 18 GHz H-pol channel */ Special values are defined as:
  0  Missing value

GAIN_VB183GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 183.31 +/ Special values are defined as:
  0  Missing value

GAIN_V23GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 23 GHz V-pol channel */ Special values are defined as:
  0  Missing value

GAIN_V36GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 36 GHz V-pol channel */ Special values are defined as:
  0  Missing value
GAIN_V89GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 89 GHz V-pol channel */ Special values are defined as:
  0  Missing value
GAIN_V166GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 166 GHz V-pol channel */ Special values are defined as:
  0  Missing value
GAIN_VA183GHZ (2-byte unsigned integer, array size: nscan2):
RS-reported gain setting for the 183.31 +/ Special values are defined as:
  0  Missing value

RSHSK_TEMP (Group in S3)

TEMP_10GHZRCVR (2-byte unsigned integer, array size: nscan2):
10 GHz Box Receiver Temperature */ Special values are defined as:
  0  Missing value
TEMP_H166GHZMXR (2-byte unsigned integer, array size: nscan2):
166 H GHz Mixer Pre-Amp Temperature */ Special values are defined as:
  0  Missing value
TEMP_18GHZRCVR (2-byte unsigned integer, array size: nscan2):
18 GHz Box Receiver Temperature */ Special values are defined as:
  0  Missing value
TEMP_V166GHZMXR (2-byte unsigned integer, array size: nscan2):
166 V GHz Mixer Pre-amp Temperature */ Special values are defined as:
  0  Missing value
TEMP_23GHZRCVR (2-byte unsigned integer, array size: nscan2):
23 GHz Box Receiver Temperature */ Special values are defined as:
  0  Missing value
TEMP_183GHZMXR (2-byte unsigned integer, array size: nscan2):
183 GHz Mixer Pre-amp Temperature */ Special values are defined as:
  0  Missing value
TEMP_36GHZRCVR (2-byte unsigned integer, array size: nscan2):
36 GHz Box Receiver Temperature */ Special values are defined as:
  0  Missing value
TEMP_H10GHZ_ND (2-byte unsigned integer, array size: nscan2):
10 GHz H-pol Noise Diode Temperature */ Special values are defined as:
  0  Missing value
TEMP_89GHZRCVR (2-byte unsigned integer, array size: nscan2):
89 GHz Box Receiver Temperature */ Special values are defined as:
  0  Missing value
5.1 1AGMI - GMI unpacked packet data

**TEMP_V10GHZ_ND** (2-byte unsigned integer, array size: nscan2):
10 GHz V-pol Noise Diode Temperature */ Special values are defined as:
  0 Missing value

**TEMP_166GHZRCVR** (2-byte unsigned integer, array size: nscan2):
166 GHz Box Receiver Temperature */ Special values are defined as:
  0 Missing value

**TEMP_H18GHZ_ND** (2-byte unsigned integer, array size: nscan2):
18 GHz H-pol Noise Diode Temperature */ Special values are defined as:
  0 Missing value

**TEMP_183GHZRCVR** (2-byte unsigned integer, array size: nscan2):
183 GHz Box Receiver Temperature */ Special values are defined as:
  0 Missing value

**TEMP_V18GHZ_ND** (2-byte unsigned integer, array size: nscan2):
18 GHz V-pol Noise Diode Temperature */ Special values are defined as:
  0 Missing value

**TEMP_RS_EPC** (2-byte unsigned integer, array size: nscan2):
Receiver Subsystem EPC Box Temperature */ Special values are defined as:
  0 Missing value

**TEMP_H36GHZ_ND** (2-byte unsigned integer, array size: nscan2):
36 GHz H-pol Noise Diode Temperature */ Special values are defined as:
  0 Missing value

**TEMP_RS_EDC** (2-byte unsigned integer, array size: nscan2):
Receiver Subsystem EDC Box Temperature */ Special values are defined as:
  0 Missing value

**TEMP_V36GHZ_ND** (2-byte unsigned integer, array size: nscan2):
36 GHz V-pol Noise Diode Temperature */ Special values are defined as:
  0 Missing value

**TEMP_FEED** (2-byte unsigned integer, array size: nscan2):
Feedhorn Assembly Temperature */ Special values are defined as:
  0 Missing value

**TEMP_89GHZ_LO** (2-byte unsigned integer, array size: nscan2):
89 GHz Local Oscillator Temperature */ Special values are defined as:
  0 Missing value

**TEMP_HL_TRAY** (2-byte unsigned integer, array size: nscan2):
Hot Load tray temperature */ Special values are defined as:
  0 Missing value

**TEMP_166GHZ_LO** (2-byte unsigned integer, array size: nscan2):
166 GHz Local Oscillator Temperature */ Special values are defined as:
  0 Missing value
**TEMP_SMASPUNHSG** (2-byte unsigned integer, array size: nscan2):
Temp SMA spun HSG */ Special values are defined as:
  0  Missing value

**TEMP_RS_MR1** (2-byte unsigned integer, array size: nscan2):
Main Reflector Temperature read by RS num 1 */ Special values are defined as:
  0  Missing value

**TEMP_H89GHZMXR** (2-byte unsigned integer, array size: nscan2):
Temp H89GHZMXR */ Special values are defined as:
  0  Missing value

**TEMP_RS_MR2** (2-byte unsigned integer, array size: nscan2):
Main Reflector Temperature read by RS num 2 */ Special values are defined as:
  0  Missing value

**TEMP_V89GHZMXR** (2-byte unsigned integer, array size: nscan2):
Temp V89GHZMXR */ Special values are defined as:
  0  Missing value

**TEMP_183GHZ_LO** (2-byte unsigned integer, array size: nscan2):
183 GHz Local Oscillator Temperature */ Special values are defined as:
  0  Missing value

**IEHSK_TEMP** (Group in S3)

**IBS_LR1_TEMP** (2-byte unsigned integer, array size: nscan2):
IBS Launch Restraint 1 temperature */ Special values are defined as:
  0  Missing value

**RS_TEMP_1** (2-byte unsigned integer, array size: nscan2):
Receiver Subsystem Temperature num 1 */ Special values are defined as:
  0  Missing value

**SCE_A_BD_TEMP** (2-byte unsigned integer, array size: nscan2):
SCE A Board Temperature */ Special values are defined as:
  0  Missing value

**MR_LR_RGHT_TEMP** (2-byte unsigned integer, array size: nscan2):
Main Reflector Right Launch Restraint Temperature */ Special values are defined as:
  0  Missing value

**SMA_BEARING_TMP** (2-byte unsigned integer, array size: nscan2):
SMA Bearing Temperature */ Special values are defined as:
  0  Missing value

**PC_BD_TEMP** (2-byte unsigned integer, array size: nscan2):
Power Controller Board Temperature */ Special values are defined as:
  0  Missing value
5.1 IAGMI - GMI unpacked packet data

LVPS_BD_TEMP (2-byte unsigned integer, array size: nscan2):
Low Voltage Power Supply Board Temperature */ Special values are defined as:
  0 Missing value

SMA_MTR_TEMP (2-byte unsigned integer, array size: nscan2):
SMA Motor Temperature */ Special values are defined as:
  0 Missing value

HL_TEMP.2 (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 2 */ Special values are defined as:
  0 Missing value

RDA_TEMP.2 (2-byte unsigned integer, array size: nscan2):
RDA Temperature num 2 */ Special values are defined as:
  0 Missing value

HL_TEMP.1 (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 1 */ Special values are defined as:
  0 Missing value

RS_TEMP.2 (2-byte unsigned integer, array size: nscan2):
Receiver Subsystem Temperature num 2 */ Special values are defined as:
  0 Missing value

SCE_B_BD_TEMP (2-byte unsigned integer, array size: nscan2):
SCE B Board Temperature */ Special values are defined as:
  0 Missing value

TEMP_CALRES.1 (2-byte unsigned integer, array size: nscan2):
ICA IE Telemetry Calibration Resistor num 1 */ Special values are defined as:
  0 Missing value

TEMP_CALRES.2 (2-byte unsigned integer, array size: nscan2):
ICA IE Telemetry Calibration Resistor num 2 */ Special values are defined as:
  0 Missing value

MR_ICA_TEMP (2-byte unsigned integer, array size: nscan2):
Main Reflector temperature read by the ICA */ Special values are defined as:
  0 Missing value

IBS_LR2_TEMP (2-byte unsigned integer, array size: nscan2):
IBS Launch Restraint 2 temperature */ Special values are defined as:
  0 Missing value

HL_TEMP.7 (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 3 */ Special values are defined as:
  0 Missing value

RDA_TEMP.1 (2-byte unsigned integer, array size: nscan2):
RDA Temperature num 1 */ Special values are defined as:
  0 Missing value
MR_LR_LEFT_TEMP (2-byte unsigned integer, array size: nscan2):
Main Reflector Left Launch Restraint Temperature */ Special values are defined as:
  0 Missing value

RDA_TEMP_3 (2-byte unsigned integer, array size: nscan2):
RDA Temperature num 3 */ Special values are defined as:
  0 Missing value

ICA_BOX_TEMP_1 (2-byte unsigned integer, array size: nscan2):
ICA Box Temperature num 1 */ Special values are defined as:
  0 Missing value

HL_TEMP_10 (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 10 */ Special values are defined as:
  0 Missing value

MR_LR_LOWR_TEMP (2-byte unsigned integer, array size: nscan2):
Main Reflector Lower Launch Restraint Temperature */ Special values are defined as:
  0 Missing value

SMA_SLPRHTR_TMP (2-byte unsigned integer, array size: nscan2):
SMA Slip Ring Heater Temperature */ Special values are defined as:
  0 Missing value

HL_TEMP_8 (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 8 */ Special values are defined as:
  0 Missing value

TEMP_CALRES_3 (2-byte unsigned integer, array size: nscan2):
ICA IE Telemetry Calibration Resistor num 3 */ Special values are defined as:
  0 Missing value

TEMP_CALRES_4 (2-byte unsigned integer, array size: nscan2):
ICA IE Telemetry Calibration Resistor num 4 */ Special values are defined as:
  0 Missing value

CSR_TEMP1 (2-byte unsigned integer, array size: nscan2):
Cold Sky Reflector Temperature */ Special values are defined as:
  0 Missing value

HL_TEMP_12 (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 12 */ Special values are defined as:
  0 Missing value

HL_TEMP_13 (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 13 */ Special values are defined as:
  0 Missing value

HL_TEMP_14 (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 14 */ Special values are defined as:
  0 Missing value
5.1 IAGMI - GMI unpacked packet data

**HL_TEMP_11** (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 11 */ Special values are defined as:
  0 Missing value

**IBS_LR3_TEMP** (2-byte unsigned integer, array size: nscan2):
IBS Launch Restraint 3 temperature */ Special values are defined as:
  0 Missing value

**HL_TEMP_9** (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 9 */ Special values are defined as:
  0 Missing value

**ROT_TEMP_SPARE** (2-byte unsigned integer, array size: nscan2):
Spare Temperature (Rotational Side) */ Special values are defined as:
  0 Missing value

**CSR_TEMP2** (2-byte unsigned integer, array size: nscan2):
Cold Sky Reflector temperature 2 */ Special values are defined as:
  0 Missing value

**CE_BD_TEMP** (2-byte unsigned integer, array size: nscan2):
CE Board Temperature */ Special values are defined as:
  0 Missing value

**ICA_BOX_TEMP_2** (2-byte unsigned integer, array size: nscan2):
ICA Box Temperature num 2 */ Special values are defined as:
  0 Missing value

**IE_BD_TEMP** (2-byte unsigned integer, array size: nscan2):
Interface Electronics Board Temperature */ Special values are defined as:
  0 Missing value

**CLR_TEMP** (2-byte unsigned integer, array size: nscan2):
CLR Temperature */ Special values are defined as:
  0 Missing value

**HL_TEMP_15** (2-byte unsigned integer, array size: nscan2):
Hot Load Temperature num 15 */ Special values are defined as:
  0 Missing value

**TEMP_CALRES_5** (2-byte unsigned integer, array size: nscan2):
ICA IE Telemetry Calibration Resistor num 5 */ Special values are defined as:
  0 Missing value

**TEMP_CALRES_6** (2-byte unsigned integer, array size: nscan2):
ICA IE Telemetry Calibration Resistor num 6 */ Special values are defined as:
  0 Missing value

**IE_TELEMETRY** (Group in S3)
IE_PASSTHRU_RSP (4-byte unsigned integer, array size: nscan2):
The response to the last pass through command. */ Special values are defined as:
  0  Missing value

IE_BLANKING_CNT (2-byte unsigned integer, array size: nscan2):
Number of blanking output pulses since last tlm cycle. */ Special values are defined as:
  0  Missing value

PWR_STAT_LR_PR (2-byte unsigned integer, array size: nscan2):
Power Controller Special values are defined as:
  0  Missing value

PWR_STAT_A (2-byte unsigned integer, array size: nscan2):
Operational Power A status */ Special values are defined as:
  0  Missing value

HTR_STAT_SMA (2-byte unsigned integer, array size: nscan2):
Operational heater status for the SMA */ Special values are defined as:
  0  Missing value

HTR_STAT_RDA (2-byte unsigned integer, array size: nscan2):
Operational heater status for the RDA */ Special values are defined as:
  0  Missing value

HTR_STAT_RS (2-byte unsigned integer, array size: nscan2):
Receiver Subsystem operational heater */ Special values are defined as:
  0  Missing value

PWR_STAT_LR_RED (2-byte unsigned integer, array size: nscan2):
Redundant Launch Restraint status */ Special values are defined as:
  0  Missing value

RS_MST_RLY_STAT (2-byte unsigned integer, array size: nscan2):
Receiver Subsytem Master Relay status */ Special values are defined as:
  0  Missing value

RDA_DEPL_STAT_4 (2-byte unsigned integer, array size: nscan2):
Reflector Deployment Assembly deployment status num 4 */ Special values are defined as:
  0  Missing value

RDA_DEPL_STAT_3 (2-byte unsigned integer, array size: nscan2):
Reflector Deployment Assembly deployment status num 3 */ Special values are defined as:
  0  Missing value

RDA_DEPL_STAT_2 (2-byte unsigned integer, array size: nscan2):
Reflector Deployment Assembly deployment status num 2 */ Special values are defined as:
  0  Missing value
**RDA_DEPL_STAT_N** (2-byte unsigned integer, array size: nscan2):
Reflector Deployment Assembly deployment status num 1 */ Special values are defined as:
- 0  Missing value

**CLR_STAT_N** (2-byte unsigned integer, array size: nscan2):
Calibration Launch Restraint Status */ Special values are defined as:
- 0  Missing value

**IBS_LR3_STAT_N** (2-byte unsigned integer, array size: nscan2):
Instrument Bay Structure Launch Restraint */ Special values are defined as:
- 0  Missing value

**MR_LR_LOWR_STAT** (2-byte unsigned integer, array size: nscan2):
Main Reflector Lower Launch Restraint */ Special values are defined as:
- 0  Missing value

**IBS_LR2_STAT_N** (2-byte unsigned integer, array size: nscan2):
Instrument Bay Structure Launch Restraint */ Special values are defined as:
- 0  Missing value

**MR_LR_LEFT_STAT** (2-byte unsigned integer, array size: nscan2):
Main Reflector Left Launch Restraint */ Special values are defined as:
- 0  Missing value

**IBS_LR1_STAT_N** (2-byte unsigned integer, array size: nscan2):
Instrument Bay Structure Launch Restraint */ Special values are defined as:
- 0  Missing value

**MR_LR_RIGHT_STAT** (2-byte unsigned integer, array size: nscan2):
Main Reflector Right Launch Restraint */ Special values are defined as:
- 0  Missing value

**IE_LATCHUP_CHAN** (2-byte unsigned integer, array size: nscan2):
Indicates the telemetry sample which observed the last */ Special values are defined as:
- 0  Missing value

**IE_LATCHUP_RETR** (2-byte unsigned integer, array size: nscan2):
Number of Retries */ Special values are defined as:
- 0  Missing value

**IE_LATCHUP_SMPS** (2-byte unsigned integer, array size: nscan2):
Number of Samples with Latchup. */ Special values are defined as:
- 0  Missing value

**IE_LATCHUP_NUM** (2-byte unsigned integer, array size: nscan2):
Number of Latchups Detected */ Special values are defined as:
- 0  Missing value

**IE_LATCHUP_FAIL** (2-byte unsigned integer, array size: nscan2):
Latchup Failure. */ Special values are defined as:
- 0  Missing value
HTR_STAT_HTLD (2-byte unsigned integer, array size: nscan2):
Operational heater status for the Hot Load */ Special values are defined as:
  0  Missing value

IE_LATCHUP_PAD (2-byte unsigned integer, array size: nscan2):
  PADDING */ Special values are defined as:
    0  Missing value

MECHANISMS (Group in S3)

SCE_LAST_CMD (4-byte unsigned integer, array size: nscan2):
Last command sent to the SCE. */ Special values are defined as:
    0  Missing value

SCE_LAST_RESPNS (4-byte unsigned integer, array size: nscan2):
Response from the SCE of the last command sent */ Special values are defined as:
    0  Missing value

SCE_CMD_CNT (4-byte unsigned integer, array size: nscan2):
Total number of cmds to the SCE. */ Special values are defined as:
    0  Missing value

SMA_RTPRB_SEC (2-byte unsigned integer, array size: nscan2):
The time since the beginning of the rate problem. 20 seconds to */ Special values are defined as:
    0  Missing value

SMA_RATE (2-byte unsigned integer, array size: nscan2):
The ICA calculated rotational rate of the SMA in integer scaled rpm */ Special values are defined as:
    0  Missing value

SCE_RATE (2-byte unsigned integer, array size: nscan2):
The SMA rotational rate reported by the SCE */ Special values are defined as:
    0  Missing value

SCE_CMD_RATE (2-byte unsigned integer, array size: nscan2):
Last rate value commanded to the SCE. Used for rate limits */ Special values are defined as:
    0  Missing value

SMA_CMD_RATE (2-byte unsigned integer, array size: nscan2):
Last rate value commanded to the SCE, converted to integer */ Special values are defined as:
    0  Missing value
RESOLVER_POS (2-byte unsigned integer, array size: nscan2):
The resolver position reported by the SCE */ Special values are defined as:
  0 Missing value

MECH.Cmd_CNT (2-byte unsigned integer, array size: nscan2):
Number of commands received by the Mechanisms CSC */ Special values are defined as:
  0 Missing value

SCE.INHIBIT (2-byte unsigned integer, array size: nscan2):
Indicator that commanding to the SCE is inhibited */ Special values are defined as:
  0 Missing value

TACH_PULSE_CNT (2-byte unsigned integer, array size: nscan2):
Array indexer for tach pulses */ Special values are defined as:
  0 Missing value

LR_ABRT_CNT (2-byte unsigned integer, array size: nscan2):
The number of launch restraint release procedures that have been */ Special values are defined as:
  0 Missing value

RAMP_ABRT_CNT (2-byte unsigned integer, array size: nscan2):
The number of ramp procedures that have been aborted. */ Special values are defined as:
  0 Missing value

OVRD.RDA.LR (2-byte unsigned integer, array size: nscan2):
Flag indicating the RDA launch restraints order protection is */ Special values are defined as:
  0 Missing value

OVRD.IBS.LR (2-byte unsigned integer, array size: nscan2):
Flag indicating the IBS launch restraints order protection is */ Special values are defined as:
  0 Missing value

SCE.PASSPROT (2-byte unsigned integer, array size: nscan2):
Flag indicating the RDA launch restraints order protection is */ Special values are defined as:
  0 Missing value

RAMP_INPROGRESS (2-byte unsigned integer, array size: nscan2):
A SMA speed modification procedure is in progress. */ Special values are defined as:
  0 Missing value

SMA_SPINNING (2-byte unsigned integer, array size: nscan2):
Indicator of whether SMA is spinning, based on speed */ Special values are defined as:
  0 Missing value
**LR_RLS_IN_PROG** (2-byte unsigned integer, array size: nscan2): Reports state of launch release fire command */ Special values are defined as:
  0   Missing value

**SCE_A_POWER** (2-byte unsigned integer, array size: nscan2): FSW status of SCE A Card power */ Special values are defined as:
  0   Missing value

**SCE_B_POWER** (2-byte unsigned integer, array size: nscan2): FSW status of SCE B Card power */ Special values are defined as:
  0   Missing value

**SCE_SELECTION** (2-byte unsigned integer, array size: nscan2): The current SCE selection setting. */ Special values are defined as:
  0   Missing value

**SMA_RATE_PROB** (2-byte unsigned integer, array size: nscan2): This field indicates the SMA is out of rate tolerances. */ Special values are defined as:
  0   Missing value

**LR_ENABLED** (2-byte unsigned integer, array size: nscan2): This tlm point indicates that one of launch restraint power buses is */ Special values are defined as:
  0   Missing value

**BILVL_IBS1** (2-byte unsigned integer, array size: nscan2): A bilevel to control the Limits Monitor CSU. */ Special values are defined as:
  0   Missing value

**BILVL_IBS2** (2-byte unsigned integer, array size: nscan2): A bilevel to control the Limits Monitor CSU. */ Special values are defined as:
  0   Missing value

**BILVL_IBS3** (2-byte unsigned integer, array size: nscan2): A bilevel to control the Limits Monitor CSU. */ Special values are defined as:
  0   Missing value

**BILVL_CAL** (2-byte unsigned integer, array size: nscan2): A bilevel to control the Limits Monitor CSU. */ Special values are defined as:
  0   Missing value

**BILVL_RDALEFT** (2-byte unsigned integer, array size: nscan2): A bilevel to control the Limits Monitor CSU. */ Special values are defined as:
  0   Missing value

**BILVL_RDARIGHT** (2-byte unsigned integer, array size: nscan2): A bilevel to control the Limits Monitor CSU. */ Special values are defined as:
  0   Missing value

**BILVL_RDALOWER** (2-byte unsigned integer, array size: nscan2): A bilevel to control the Limits Monitor CSU. */ Special values are defined as:
  0   Missing value
SMPL_INFO (Group in S3)

SMPL_INFO_VALID (2-byte unsigned integer, array size: nscan2):
Sample Table Valid */ Special values are defined as:
  0 Missing value

EARTH_10G_STRT (2-byte unsigned integer, array size: nscan2):
Earth viewing start (10GHz) */ Special values are defined as:
  0 Missing value

EARTH_10G_NUM (2-byte unsigned integer, array size: nscan2):
Earth viewing samples (10GHz) */ Special values are defined as:
  0 Missing value

HLOAD_10G_STRT (2-byte unsigned integer, array size: nscan2):
Hot Load start (10GHz) */ Special values are defined as:
  0 Missing value

HLOAD_10G_NUM (2-byte unsigned integer, array size: nscan2):
Hot Load samples (10GHz) */ Special values are defined as:
  0 Missing value

CSKY_10G_STRT (2-byte unsigned integer, array size: nscan2):
Cold Sky start (10GHz) */ Special values are defined as:
  0 Missing value

CSKY_10G_NUM (2-byte unsigned integer, array size: nscan2):
Cold Sky samples (10GHz) */ Special values are defined as:
  0 Missing value

EARTH_18G_STRT (2-byte unsigned integer, array size: nscan2):
Earth viewing start (18GHz) */ Special values are defined as:
  0 Missing value

EARTH_18G_NUM (2-byte unsigned integer, array size: nscan2):
Earth viewing samples (18GHz) */ Special values are defined as:
  0 Missing value

HLOAD_18G_STRT (2-byte unsigned integer, array size: nscan2):
Hot Load start (18GHz) */ Special values are defined as:
  0 Missing value

HLOAD_18G_NUM (2-byte unsigned integer, array size: nscan2):
Hot Load samples (18GHz) */ Special values are defined as:
  0 Missing value

CSKY_18G_STRT (2-byte unsigned integer, array size: nscan2):
Cold Sky start (18GHz) */ Special values are defined as:
  0 Missing value
**CSKY_18G_NUM** (2-byte unsigned integer, array size: nscan2):
Cold Sky samples (18GHz) */ Special values are defined as:
0 Missing value

**EARTH_23G_STRT** (2-byte unsigned integer, array size: nscan2):
Earth viewing start (23GHz) */ Special values are defined as:
0 Missing value

**EARTH_23G_NUM** (2-byte unsigned integer, array size: nscan2):
Earth viewing samples (23GHz) */ Special values are defined as:
0 Missing value

**HLOAD_23G_STRT** (2-byte unsigned integer, array size: nscan2):
Hot Load start (23GHz) */ Special values are defined as:
0 Missing value

**HLOAD_23G_NUM** (2-byte unsigned integer, array size: nscan2):
Hot Load samples (23GHz) */ Special values are defined as:
0 Missing value

**CSKY_23G_STRT** (2-byte unsigned integer, array size: nscan2):
Cold Sky start (23GHz) */ Special values are defined as:
0 Missing value

**CSKY_23G_NUM** (2-byte unsigned integer, array size: nscan2):
Cold Sky samples (23GHz) */ Special values are defined as:
0 Missing value

**EARTH_36G_STRT** (2-byte unsigned integer, array size: nscan2):
Earth viewing start (36GHz) */ Special values are defined as:
0 Missing value

**EARTH_36G_NUM** (2-byte unsigned integer, array size: nscan2):
Earth viewing samples (36GHz) */ Special values are defined as:
0 Missing value

**HLOAD_36G_STRT** (2-byte unsigned integer, array size: nscan2):
Hot Load start (36GHz) */ Special values are defined as:
0 Missing value

**HLOAD_36G_NUM** (2-byte unsigned integer, array size: nscan2):
Hot Load samples (36GHz) */ Special values are defined as:
0 Missing value

**CSKY_36G_STRT** (2-byte unsigned integer, array size: nscan2):
Cold Sky start (36GHz) */ Special values are defined as:
0 Missing value

**CSKY_36G_NUM** (2-byte unsigned integer, array size: nscan2):
Cold Sky samples (36GHz) */ Special values are defined as:
0 Missing value
5.1 1AGMI - GMI unpacked packet data

**EARTH\_89G\_STRT** (2-byte unsigned integer, array size: nscan2):
Earth viewing start (89GHz) */ Special values are defined as:
0 Missing value

**EARTH\_89G\_NUM** (2-byte unsigned integer, array size: nscan2):
Earth viewing samples (89GHz) */ Special values are defined as:
0 Missing value

**HLOAD\_89G\_STRT** (2-byte unsigned integer, array size: nscan2):
Hot Load start (89GHz) */ Special values are defined as:
0 Missing value

**HLOAD\_89G\_NUM** (2-byte unsigned integer, array size: nscan2):
Hot Load samples (89GHz) */ Special values are defined as:
0 Missing value

**CSKY\_89G\_STRT** (2-byte unsigned integer, array size: nscan2):
Cold Sky start (89GHz) */ Special values are defined as:
0 Missing value

**CSKY\_89G\_NUM** (2-byte unsigned integer, array size: nscan2):
Cold Sky samples (89GHz) */ Special values are defined as:
0 Missing value

**EARTH\_166G\_STRT** (2-byte unsigned integer, array size: nscan2):
Earth viewing start (166GHz) */ Special values are defined as:
0 Missing value

**EARTH\_166G\_NUM** (2-byte unsigned integer, array size: nscan2):
Earth viewing samples (166GHz) */ Special values are defined as:
0 Missing value

**HLOAD\_166G\_STRT** (2-byte unsigned integer, array size: nscan2):
Hot Load start (166GHz) */ Special values are defined as:
0 Missing value

**HLOAD\_166G\_NUM** (2-byte unsigned integer, array size: nscan2):
Hot Load samples (166GHz) */ Special values are defined as:
0 Missing value

**CSKY\_166G\_STRT** (2-byte unsigned integer, array size: nscan2):
Cold Sky start (166GHz) */ Special values are defined as:
0 Missing value

**CSKY\_166G\_NUM** (2-byte unsigned integer, array size: nscan2):
Cold Sky samples (166GHz) */ Special values are defined as:
0 Missing value

**EARTH\_183G\_STRT** (2-byte unsigned integer, array size: nscan2):
Earth viewing start (183GHz) */ Special values are defined as:
0 Missing value
**Earth Viewing Samples (183GHz)** */ Special values are defined as:
- **0** Missing value

**Hot Load Start (183GHz)** */ Special values are defined as:
- **0** Missing value

**Hot Load Samples (183GHz)** */ Special values are defined as:
- **0** Missing value

**Cold Sky Start (183GHz)** */ Special values are defined as:
- **0** Missing value

**Cold Sky Samples (183GHz)** */ Special values are defined as:
- **0** Missing value

### S4 (Swath)

**S4_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S4)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- **-9999** Missing value

**Month** (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
- **-99** Missing value

**DayOfMonth** (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
- **-99** Missing value

**Hour** (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
- **-99** Missing value
5.1 IAGMI - GMI unpacked packet data

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

Millisecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

ephemerisUsed (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
255 Missing value

Latitude (4-byte float, array size: npixelfr x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixelfr x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

incidenceAngle (4-byte float, array size: npixelfr x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

fullRotation (2-byte unsigned integer, array size: nchannel1 x npixelfr x nscan2):

Full rotation counts.

Special values are defined as:
0 Missing value
fullRotBlanking (1-byte char, array size: VH x npixelfr x nscan2):

Full rotation blanking counts.

Special values are defined as:
0  Missing value

S5 (Swath)

S5_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S5)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value
DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

ephemerisUsed (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
255 Missing value

Latitude (4-byte float, array size: npixelfr x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixelfr x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

incidenceAngle (4-byte float, array size: npixelfr x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel
location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

fullRotation (2-byte unsigned integer, array size: nchannel2 x npixelfr x nscan2):

Full rotation counts.

Special values are defined as:
0 Missing value

C Structure Header file:

```c
#define _TK_1AGMI_H_
#define _TK_1AGMI_H_

#define _L1AGMI_S5_
#define _L1AGMI_S5_

typedef struct {
```
SCANTIME ScanTime;
unsigned char ephemerisUsed[10];
float Latitude[500];
float Longitude[500];
float incidenceAngle[500];
unsigned short fullRotation[500][4];
} L1AGMI_S5;

#endif

#ifndef _L1AGMI_S4_
#define _L1AGMI_S4_

typedef struct {
    SCANTIME ScanTime;
    unsigned char ephemerisUsed[10];
    float Latitude[500];
    float Longitude[500];
    float incidenceAngle[500];
    unsigned short fullRotation[500][9];
    unsigned char fullRotBlanking[500][2];
} L1AGMI_S4;
#endif

#ifndef _L1AGMI_S3_SMPL_INFO_
#define _L1AGMI_S3_SMPL_INFO_

typedef struct {
    unsigned short SMPL_INFO_VALID;
    unsigned short EARTH_10G_STRT;
    unsigned short EARTH_10G_NUM;
    unsigned short HLOAD_10G_STRT;
    unsigned short HLOAD_10G_NUM;
    unsigned short CSKY_10G_STRT;
    unsigned short CSKY_10G_NUM;
    unsigned short EARTH_18G_STRT;
    unsigned short EARTH_18G_NUM;
    unsigned short HLOAD_18G_STRT;
    unsigned short HLOAD_18G_NUM;
    unsigned short CSKY_18G_STRT;
    unsigned short CSKY_18G_NUM;
    unsigned short EARTH_23G_STRT;
    unsigned short EARTH_23G_NUM;
unsigned short EARTH_23G_NUM;
unsigned short HLOAD_23G_NUM;
unsigned short CSKY_23G_NUM;
unsigned short EARTH_36G_NUM;
unsigned short HLOAD_36G_NUM;
unsigned short CSKY_36G_NUM;
unsigned short EARTH_89G_NUM;
unsigned short HLOAD_89G_NUM;
unsigned short CSKY_89G_NUM;
unsigned short EARTH_166G_NUM;
unsigned short HLOAD_166G_NUM;
unsigned short CSKY_166G_NUM;
unsigned short EARTH_183G_NUM;
unsigned short HLOAD_183G_NUM;
unsigned short CSKY_183G_NUM;

} L1AGMI_S3_SMPL_INFO;

#endif

#ifndef _L1AGMI_S3_MECHANISMS_
#define _L1AGMI_S3_MECHANISMS_

typedef struct {
    unsigned int SCE_LAST_CMD;
    unsigned int SCE_LAST_RESPNS;
    unsigned int SCE_CMD_CNT;
    unsigned short SMA_RTPRB_SEC;
    unsigned short SMA_RATE;
    unsigned short SCE_RATE;

以外はコードが続く。
unsigned short SCE_CMD_RATE;
unsigned short SMA_CMD_RATE;
unsigned short RESOLVER_POS;
unsigned short MECH_CMD_CNT;
unsigned short SCE_INHIBIT;
unsigned short TACH_PULSE_CNT;
unsigned short LR_ABRT_CNT;
unsigned short RAMP_ABRT_CNT;
unsigned short OVRD_RDA_LR;
unsigned short OVRD_IBS_LR;
unsigned short SCE_PASSPROT;
unsigned short RAMP_INPROGRESS;
unsigned short SMA_SPINNING;
unsigned short LR_RLS_IN_PROG;
unsigned short SCE_A_POWER;
unsigned short SCE_B_POWER;
unsigned short SCE_SELECTION;
unsigned short SMA_RATE_PROB;
unsigned short LR_ENABLED;
unsigned short BILVL_IBS1;
unsigned short BILVL_IBS2;
unsigned short BILVL_IBS3;
unsigned short BILVL_CAL;
unsigned short BILVL_RDALeft;
unsigned short BILVL_RDARIGHT;
unsigned short BILVL_RDALOWER;
}; L1AGMI_S3_MECHANISMS;

#endif

#ifndef _L1AGMI_S3_IE_TELEMETRY_
define _L1AGMI_S3_IE_TELEMETRY_

typedef struct {
    unsigned int IE_PASSTHRU_RSP;
    unsigned short IE_BLANKING_CNT;
    unsigned short PWR_STAT_LR_PR;
    unsigned short PWR_STAT_A;
    unsigned short HTR_STAT_SMA;
    unsigned short HTR_STAT_RDA;
    unsigned short HTR_STAT_RS;
    unsigned short PWR_STAT_LR_RED;
    unsigned short RS_MST_RLY_STAT;
}
unsigned short RDA_Depl_Stat_4;
unsigned short RDA_Depl_Stat_3;
unsigned short RDA_Depl_Stat_2;
unsigned short RDA_Depl_Stat_1;
unsigned short CLR_STAT_N;
unsigned short IBS_Lr3_Stat_N;
unsigned short Mr_Lr_Lowr_Stat;
unsigned short IBS_Lr2_Stat_N;
unsigned short Mr_Lr_Left_Stat;
unsigned short IBS_Lr1_Stat_N;
unsigned short Mr_Lr_Rght_Stat;
unsigned short IE_Latchup_Chn;
unsigned short IE_Latchup_ReTr;
unsigned short Mr_Lr_Lowr_Stat;
unsigned short IE_Latchup_Smps;
unsigned short IE_Latchup_Num;
unsigned short IE_Latchup_Fail;
unsigned short Htr_Stat_Htld;
unsigned short IE_Latchup_pad;
} L1AGMI_S3_IE_Telemetry;

#endif

#ifndef _L1AGMI_S3_IEHSK_TEMP_
define _L1AGMI_S3_IEHSK_TEMP_

typedef struct {
    unsigned short IBS_Lr1_Temp;
    unsigned short Rs_Temp_1;
    unsigned short Sce_A_Bd_Temp;
    unsigned short Mr_Lr_Rght_Temp;
    unsigned short SmA_Bearing_Tmp;
    unsigned short Pc_Bd_Temp;
    unsigned short Lvps_Bd_Temp;
    unsigned short SmA_Mtr_Temp;
    unsigned short Hl_Temp_2;
    unsigned short Rda_Temp_2;
    unsigned short Hl_Temp_1;
    unsigned short Rs_Temp_2;
    unsigned short Sce_B_Bd_Temp;
    unsigned short Temp_Calres_1;
    unsigned short Temp_Calres_2;
    unsigned short Mr_ICA_Temp;
    unsigned short IBS_Lr2_Temp;
} L1AGMI_S3_IEHSK_TEMP;
unsigned short HL_TEMP_7;
unsigned short RDA_TEMP_1;
unsigned short MR_LR_LEFT_TEMP;
unsigned short RDA_TEMP_3;
unsigned short ICA_BOX_TEMP_1;
unsigned short HL_TEMP_10;
unsigned short MR_LR_LOWR_TEMP;
unsigned short SMA_SLPRHTR_TMP;
unsigned short HL_TEMP_8;
unsigned short TEMP_CALRES_3;
unsigned short TEMP_CALRES_4;
unsigned short CSR_TEMP1;
unsigned short HL_TEMP_12;
unsigned short HL_TEMP_13;
unsigned short HL_TEMP_14;
unsigned short HL_TEMP_11;
unsigned short IBS_LR3_TEMP;
unsigned short HL_TEMP_9;
unsigned short ROT_TEMP_SPARE;
unsigned short CSR_TEMP2;
unsigned short CE_BD_TEMP;
unsigned short ICA_BOX_TEMP_2;
unsigned short IE_BD_TEMP;
unsigned short CLR_TEMP;
unsigned short HL_TEMP_15;
unsigned short TEMP_CALRES_5;
unsigned short TEMP_CALRES_6;
} L1AGMI_S3_IEHSK_TEMP;

#endif

#ifndef _L1AGMI_S3_RSHSK_TEMP_
define _L1AGMI_S3_RSHSK_TEMP_

typedef struct {
    unsigned short TEMP_10GHZRCVR;
    unsigned short TEMP_H166GHZMXR;
    unsigned short TEMP_18GHZRCVR;
    unsigned short TEMP_V166GHZMXR;
    unsigned short TEMP_23GHZRCVR;
    unsigned short TEMP_183GHZMXR;
    unsigned short TEMP_36GHZRCVR;
    unsigned short TEMP_H10GHZ_ND;
}
unsigned short TEMP_89GHZRCVR;
unsigned short TEMP_V10GHZ_ND;
unsigned short TEMP_166GHZRCVR;
unsigned short TEMP_H18GHZ_ND;
unsigned short TEMP_183GHZRCVR;
unsigned short TEMP_V18GHZ_ND;
unsigned short TEMP_RS_EPC;
unsigned short TEMP_H36GHZ_ND;
unsigned short TEMP_RS_EDC;
unsigned short TEMP_V36GHZ_ND;
unsigned short TEMP_FEED;
unsigned short TEMP_89GHZ_L0;
unsigned short TEMP_HL_TRAY;
unsigned short TEMP_166GHZ_L0;
unsigned short TEMP_SMASPUNHSG;
unsigned short TEMP_RS_MR1;
unsigned short TEMP_H89GHZMXR;
unsigned short TEMP_RS_MR2;
unsigned short TEMP_V89GHZMXR;
unsigned short TEMP_183GHZ_L0;
} L1AGMI_S3_RSHSK_TEMP;

#endif

#ifndef _L1AGMI_S3_RSHSK_GAIN_
#define _L1AGMI_S3_RSHSK_GAIN_

typedef struct {
  unsigned short RESERVED2;
  unsigned short GAIN_V10GHZ;
  unsigned short GAIN_H36GHZ;
  unsigned short GAIN_H89GHZ;
  unsigned short GAIN_H10GHZ;
  unsigned short GAIN_H166GHZ;
  unsigned short GAIN_V18GHZ;
  unsigned short GAIN_H18GHZ;
  unsigned short GAIN_VB183GHZ;
  unsigned short GAIN_V23GHZ;
  unsigned short GAIN_V36GHZ;
  unsigned short GAIN_V89GHZ;
  unsigned short GAIN_V166GHZ;
  unsigned short GAIN_VA183GHZ;
} L1AGMI_S3_RSHSK_GAIN;
#ifndef _L1AGMI_S3_RSHSK_SAMPL_INFO_
define _L1AGMI_S3_RSHSK_SAMPL_INFO_

typedef struct {
    unsigned short SMPOFFST_10GHZ;
    unsigned short SMPOFFST_18GHZ;
    unsigned short SMPOFFST_23GHZ;
    unsigned short SMPOFFST_36GHZ;
    unsigned short SMPOFFST_89GHZ;
    unsigned short SMPOFFST_166GHZ;
    unsigned short SMPOFFST_183GHZ;
    unsigned short NUMSMPLS_10GHZ;
    unsigned short NUMSMPLS_18GHZ;
    unsigned short NUMSMPLS_23GHZ;
    unsigned short NUMSMPLS_36GHZ;
    unsigned short NUMSMPLS_89GHZ;
    unsigned short NUMSMPLS_166GHZ;
    unsigned short NUMSMPLS_183GHZ;
} L1AGMI_S3_RSHSK_SAMPL_INFO;
#endif

#ifndef _L1AGMI_S3_RSHSK_STATUS_
define _L1AGMI_S3_RSHSK_STATUS_

typedef struct {
    unsigned short RSST_SCI_ADC_LP;
    unsigned short RSST_HSK_ADC_LP;
    unsigned short RSST_SAMP_OVLP;
    unsigned short RSST_10GHZ_RLY;
    unsigned short RSST_18GHZ_RLY;
    unsigned short RSST_23GHZ_RLY;
    unsigned short RSST_36GHZ_RLY;
    unsigned short RSST_89GHZ_RLY;
    unsigned short RSST_166GHZ_RLY;
    unsigned short RSST_183GHZ_RLY;
    unsigned short RSST_INVLD_CMD;
    unsigned short RSST_CMD_AFTER;
    unsigned short NDIODE_MODE;
    unsigned short RSST_NDIODE_ST;
} L1AGMI_S3_RSHSK_STATUS;
#endif
typedef struct {
    unsigned int TACH_SECS_16;
    unsigned short TACH_SUBS_16;
    unsigned int TACH_SECS_17;
    unsigned short TACH_SUBS_17;
    unsigned int TACH_SECS_18;
    unsigned short TACH_SUBS_18;
    unsigned int TACH_SECS_19;
    unsigned short TACH_SUBS_19;
    unsigned int TACH_SECS_20;
    unsigned short TACH_SUBS_20;
    unsigned int TACH_SECS_21;
    unsigned short TACH_SUBS_21;
    unsigned int TACH_SECS_22;
    unsigned short TACH_SUBS_22;
    unsigned int TACH_SECS_23;
    unsigned short TACH_SUBS_23;
    unsigned int TACH_SECS_24;
    unsigned short TACH_SUBS_24;
    unsigned int TACH_SECS_25;
    unsigned short TACH_SUBS_25;
    unsigned int TACH_SECS_26;
    unsigned short TACH_SUBS_26;
    unsigned int TACH_SECS_27;
} L1AGMI_S3_RSHSK_STATUS;

#endif

#ifndef _L1AGMI_S3_SYNCH_STAMPS2_
#define _L1AGMI_S3_SYNCH_STAMPS2_

unsigned short NDIODE10GHZSNUM;
unsigned short RESERVED1;
unsigned short RS_CALRES_1;
unsigned short BATC_CALRES_1;
unsigned short RS_CALRES_2;
unsigned short BATC_CALRES_2;
unsigned short RS_EPC_ISENS;
unsigned short RS_EPC_5V;
unsigned short RS_EPC_7V;
unsigned short RS_EPC_POS12V;
unsigned short RS_EPC_NEG12V;
unsigned short RS_EPC_15V;
} L1AGMI_S3_RSHSK_STATUS;

#endif

#ifndef _L1AGMI_S3_SYNCH_STAMPS2_
#define _L1AGMI_S3_SYNCH_STAMPS2_

typedef struct {
    unsigned int TACH_SECS_16;
    unsigned short TACH_SUBS_16;
    unsigned int TACH_SECS_17;
    unsigned short TACH_SUBS_17;
    unsigned int TACH_SECS_18;
    unsigned short TACH_SUBS_18;
    unsigned int TACH_SECS_19;
    unsigned short TACH_SUBS_19;
    unsigned int TACH_SECS_20;
    unsigned short TACH_SUBS_20;
    unsigned int TACH_SECS_21;
    unsigned short TACH_SUBS_21;
    unsigned int TACH_SECS_22;
    unsigned short TACH_SUBS_22;
    unsigned int TACH_SECS_23;
    unsigned short TACH_SUBS_23;
    unsigned int TACH_SECS_24;
    unsigned short TACH_SUBS_24;
    unsigned int TACH_SECS_25;
    unsigned short TACH_SUBS_25;
    unsigned int TACH_SECS_26;
    unsigned short TACH_SUBS_26;
    unsigned int TACH_SECS_27;
typedef struct {
    unsigned int IDX_PULSE_SECS;
    unsigned short IDX_PULSE_SUBS;
    unsigned int TACH_SECS_00;
    unsigned short TACH_SUBS_00;
    unsigned int TACH_SECS_01;
    unsigned short TACH_SUBS_01;
    unsigned int TACH_SECS_02;
    unsigned short TACH_SUBS_02;
    unsigned int TACH_SECS_03;
    unsigned short TACH_SUBS_03;
    unsigned int TACH_SECS_04;
    unsigned short TACH_SUBS_04;
    unsigned int TACH_SECS_05;
    unsigned short TACH_SUBS_05;
    unsigned int TACH_SECS_06;
    unsigned short TACH_SUBS_06;
    unsigned int TACH_SECS_07;
    unsigned short TACH_SUBS_07;
    unsigned int TACH_SECS_08;
    unsigned short TACH_SUBS_08;
    unsigned int TACH_SECS_09;
    unsigned short TACH_SUBS_09;
    unsigned int TACH_SECS_10;
    unsigned short TACH_SUBS_10;
} L1AGMI_S3_SYNCH_STAMPS2;

#endif

#ifndef _L1AGMI_S3_SYNCH_STAMPS_
#define _L1AGMI_S3_SYNCH_STAMPS_

unsigned short TACH_SUBS_27;
unsigned int TACH_SECS_28;
unsigned short TACH_SUBS_28;
unsigned int TACH_SECS_29;
unsigned short TACH_SUBS_29;
unsigned int TACH_SECS_30;
unsigned short TACH_SUBS_30;
unsigned int TACH_SECS_31;
unsigned short TACH_SUBS_31;
unsigned int SCAN_COMPL_SECS;
unsigned short SCAN_COMPL_SUBS;

} L1AGMI_S3_SYNCH_STAMPS2;
unsigned int TACH_SECS_11;
unsigned short TACH_SUBS_11;
unsigned int TACH_SECS_12;
unsigned short TACH_SUBS_12;
unsigned int TACH_SECS_13;
unsigned short TACH_SUBS_13;
unsigned int TACH_SECS_14;
unsigned short TACH_SUBS_14;
unsigned int TACH_SECS_15;
unsigned short TACH_SUBS_15;
} L1AGMI_S3_SYNCH_STAMPS;

#endif

#ifndef _L1AGMI_S3_RS_INFO_
#define _L1AGMI_S3_RS_INFO_

typedef struct {
    unsigned short RS_POWERED;
    unsigned short RS_ENABLED;
    unsigned short RS_MST_RLY;
    unsigned short RS_10GHZ_RLY;
    unsigned short RS_18GHZ_RLY;
    unsigned short RS_23GHZ_RLY;
    unsigned short RS_36GHZ_RLY;
    unsigned short RS_89GHZ_RLY;
    unsigned short RS_166GHZ_RLY;
    unsigned short RS_183GHZ_RLY;
    unsigned short RS_DQ_MISSING;
    unsigned short RS_DQ_EXTRAS;
    unsigned short RS_DQ_DUPES;
    unsigned short RS_LAST_REV;
    unsigned short RS_DQSAME_REV;
    unsigned short RS_DQBAD_REVS;
    unsigned short RS_DQ_PAR_ERR;
    unsigned short RS_DQ_CLK_ERR;
    unsigned short RS_DQ_PKT_ERR;
    unsigned short RS_DQ_TLM_ERR;
    unsigned short RS_DQBAD_CONF;
    unsigned short RS_DQ_CAL_LIM;
    unsigned short BLK_STATE;
    unsigned short BLK_SIDE;
    unsigned short BLK_DELAY;
} L1AGMI_S3_RS_INFO;

#endif

#define _L1AGMI_S3_RS_INFO_
unsigned short BLK_DURATION;
unsigned short RS_HSK_SIZE;
unsigned int RS_PAR_ERR_CNT;
unsigned int RS_SCAN_CNT;
unsigned int SAMPLE_TBL_VER;
unsigned int SMPL_TBL;
unsigned short RS_SC_SIZE;
unsigned short GSDR_SIZE;
unsigned short GSDR_LEFT;
unsigned short GSDR_B_POP_IDX;
unsigned short GSDR_B_PUSH_IDX;
unsigned short GSDR_APID;
unsigned short PKT_STATE;
unsigned short OVRD_RS_PWR;
unsigned short OVRD_SMA_SPIN;
unsigned short OVRD_PASS_RS;
} L1AGMI_S3_RS_INFO;

#endif

#ifndef _L1AGMI_S3_SENSOR_INFO_
#define _L1AGMI_S3_SENSOR_INFO_

typedef struct {
    unsigned int KEEP_ALIVE_CNT;
    unsigned short FPGA_RST_REASON;
    unsigned short CRASH_REASON;
    unsigned short VERSION_MIN;
    unsigned short VERSION_MAJ;
    unsigned short FPGA_MODE;
    unsigned short ERR_HDL_FAILURE;
    unsigned short RESET_REASON;
    unsigned short BOOT_BANK;
    unsigned short CURRENT_BANK;
    unsigned short EDAC_ENABLE;
    unsigned short WDOG_ENABLE;
    unsigned short SC_1HZ_REF;
    unsigned short SCE_FORCE_SEL;
    unsigned short RS_1MHZ_REF;
    unsigned short RS_SCAN_START;
    unsigned short FPGA_IE_RX_EN;
    unsigned short FPGA_RS_RX_EN;
    unsigned short FPGA_SCE_RX_EN;
} L1AGMI_S3_SENSOR_INFO;
unsigned short EEPROM_BUSY;
unsigned short RS_TLM_PROG;
unsigned short SCE_A_ACTIVE;
unsigned short SCE_A_RLY;
unsigned short SCE_B_ACTIVE;
unsigned short SCE_B_RLY;
unsigned short IE_TLM_PROG;
unsigned short SCE_RSP_PROG;
unsigned short RS_CLK_ERR;
unsigned short RS_PKT_ERR;
unsigned short RS_TLM_ERR;
unsigned short SCE_RSP_RDY;
unsigned short IE_PKT_ERR;
unsigned short IE_CMD_ERR;
unsigned short IE_RSP_ERR;
unsigned short IE_TLM_ERR;
unsigned short FPGA_ACCESS_ERR;
} L1AGMI_S3_SENSOR_INFO;

#endif

#ifndef _L1AGMI_S3_GSDR_TIME_
#define _L1AGMI_S3_GSDR_TIME_

typedef struct {
    unsigned int G_TC_PULSE_SECS;
    unsigned short G_TC_PULSE_SUBS;
    unsigned int G_TCU_SECS;
    unsigned int G_TCU_SUBS;
    unsigned int G_TCF_SC_SECS;
    unsigned int G_TCF_SC_SUBSEC;
    unsigned int G_TCF_SECS;
    unsigned int G_TCF_SUBSECS;
    unsigned int G_TCF_SUBSECS;
    unsigned short G_TCF_SIGN;
    unsigned short G_TCF_LEAP;
    unsigned int GPS_TCU_SECS;
    unsigned int GPS_TCU_SUBS;
} L1AGMI_S3_GSDR_TIME;

#endif

#ifndef _PRIMARYHEADER_
#define _PRIMARYHEADER_

#define _GRDAGMI_S3_GSDR_TIME_
typedef struct {
    signed char version;
    signed char type;
    signed char secHeaderFlag;
    short APID;
    signed char sequenceFlag;
    short packetSequenceCount;
    unsigned short packetLength;
} PRIMARYHEADER;

#endif

#ifndef _L1AGMI_S3_GMI_TEMPERATURES_
#define _L1AGMI_S3_GMI_TEMPERATURES_

typedef struct {
    float timeOffset;
    unsigned short apid;
    unsigned short SMA_PT_TEMP;
    unsigned short ICA_PT_TEMP;
    unsigned short RS_PT_TEMP;
    unsigned short STAT_PT_TEMP;
    unsigned short MR_PT_TEMP;
} L1AGMI_S3_GMI_TEMPERATURES;
#endif

#ifndef _L1AGMI_S3_TORQUE_BAR_
#define _L1AGMI_S3_TORQUE_BAR_

typedef struct {
    float timeOffset;
    unsigned short Vx;
    unsigned short Vy;
    unsigned short Vz;
} L1AGMI_S3_TORQUE_BAR;
#endif

#ifndef _L1AGMI_S3_TAM2_
#define _L1AGMI_S3_TAM2_

typedef struct {
    float timeOffset;
    unsigned short Vx;
    unsigned short Vy;
    unsigned short Vz;
} L1AGMI_S3_TAM2;

#endif

#ifndef _L1AGMI_S3_TAM1_
#define _L1AGMI_S3_TAM1_

typedef struct {
    float timeOffset;
    unsigned short Vx;
    unsigned short Vy;
    unsigned short Vz;
} L1AGMI_S3_TAM1;

#endif

#ifndef _L1AGMI_S3_
#define _L1AGMI_S3_

typedef struct {
    SCANTIME ScanTime;
    unsigned char ephemerisUsed[10];
    float Latitude;
    float Longitude;
    L1AGMI_S3_TAM1 TAM1;
    L1AGMI_S3_TAM2 TAM2;
    L1AGMI_S3_TORQUE_BAR TORQUE_BAR;
    L1AGMI_S3_GMI_TEMPERATURES GMI_TEMPERATURES;
    PRIMARYHEADER primaryHeader;
    unsigned int instrTimeSeconds;
    unsigned short instrTimeSubseconds;
    signed char numPacketSegments;
    signed char spare;
    short RDRversion;
    L1AGMI_S3_GSDR_TIME GSDR_TIME;
    L1AGMI_S3_SENSOR_INFO SENSOR_INFO;
    L1AGMI_S3_RS_INFO RS_INFO;
    L1AGMI_S3_SYNCH_STAMPS SYNCH_STAMPS;
} 1AGMI - GMI unpacked packet data 153
typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1AGMI_S2_SUNDATA;

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
} L1AGMI_S2_SCANSTATUS;
double FractionalGranuleNumber;
} L1AGMI_S2_SCANSTATUS;

#endif

#ifndef _L1AGMI_S2_
define _L1AGMI_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    L1AGMI_S2_SCANSTATUS scanStatus;
    unsigned char ephemerisUsed[10];
    NAVIGATION navigation;
    L1AGMI_S2_SUNDATA sunData;
    float incidenceAngle[221];
    float satAzimuthAngle[221];
    float solarZenAngle[221];
    float solarAzimuthAngle[221];
    float sunGlintAngle[221];
    float moonVectorInstFrame[3];
    unsigned short earthView[221][4];
    unsigned short hotLoad[221][4];
    unsigned short coldSky[221][4];
} L1AGMI_S2;
#endif

#ifndef _L1AGMI_S1_SUNDATA_
define _L1AGMI_S1_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1AGMI_S1_SUNDATA;
typedef struct {
    float scPos[3];
    float scVel[3];
    float sclat;
    float sclon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1AGMI_S1_SCANSTATUS;

```c
#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L1AGMI_S1_
#define _L1AGMI_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    L1AGMI_S1SCANSTATUS scanStatus;
    unsigned char ephemerisUsed[10];
    NAVIGATION navigation;
    L1AGMI_S1SUNDATA sunData;
    float incidenceAngle[221];
    float satAzimuthAngle[221];
    float solarZenAngle[221];
    float solarAzimuthAngle[221];
    float sunGlintAngle[221];
    float moonVectorInstFrame[3];
    unsigned short earthView[221][9];
    unsigned short hotLoad[221][9];
    unsigned short coldSky[221][9];
    unsigned char earthViewBlanking[221][2];
    unsigned char hotLoadBlanking[221][2];
    unsigned char coldSkyBlanking[221][2];
} L1AGMI_S1;
```

#endif

#ifndef _L1AGMI_SWATHS_
define _L1AGMI_SWATHS_

typedef struct {
    L1AGMI_S1 S1;
    L1AGMI_S2 S2;
    L1AGMI_S3 S3;
    L1AGMI_S4 S4;
    L1AGMI_S5 S5;
} L1AGMI_SWATHS;
#endif

#ifndef _L1AGMI_GMI1AHEADER_
define _L1AGMI_GMI1AHEADER_

typedef struct {
    unsigned short sampleRangeFile[7][6];
} L1AGMI_GMI1AHEADER;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L1AGMI_S5/
    RECORD /SCANTIME/ ScanTime
    CHARACTER ephemerisUsed(10)
    REAL*4 Latitude(500)
    REAL*4 Longitude(500)
    REAL*4 incidenceAngle(500)
    INTEGER*2 fullRotation(4,500)
END STRUCTURE

STRUCTURE /L1AGMI_S4/
    RECORD /SCANTIME/ ScanTime
    CHARACTER ephemerisUsed(10)
    REAL*4 Latitude(500)
    REAL*4 Longitude(500)
    REAL*4 incidenceAngle(500)
5.1 1AGMI - GMI unpacked packet data

INTEGER*2 fullRotation(9,500)
CHARACTER fullRotBlanking(2,500)
END STRUCTURE

STRUCTURE /L1AGMI_S3_SMPL_INFO/
  INTEGER*2 SMPL_INFO_VALID
  INTEGER*2 EARTH_10G_STRT
  INTEGER*2 EARTH_10G_NUM
  INTEGER*2 HLOAD_10G_STRT
  INTEGER*2 HLOAD_10G_NUM
  INTEGER*2 CSKY_10G_STRT
  INTEGER*2 CSKY_10G_NUM
  INTEGER*2 EARTH_18G_STRT
  INTEGER*2 EARTH_18G_NUM
  INTEGER*2 HLOAD_18G_STRT
  INTEGER*2 HLOAD_18G_NUM
  INTEGER*2 CSKY_18G_STRT
  INTEGER*2 CSKY_18G_NUM
  INTEGER*2 EARTH_23G_STRT
  INTEGER*2 EARTH_23G_NUM
  INTEGER*2 HLOAD_23G_STRT
  INTEGER*2 HLOAD_23G_NUM
  INTEGER*2 CSKY_23G_STRT
  INTEGER*2 CSKY_23G_NUM
  INTEGER*2 EARTH_36G_STRT
  INTEGER*2 EARTH_36G_NUM
  INTEGER*2 HLOAD_36G_STRT
  INTEGER*2 HLOAD_36G_NUM
  INTEGER*2 CSKY_36G_STRT
  INTEGER*2 CSKY_36G_NUM
  INTEGER*2 EARTH_89G_STRT
  INTEGER*2 EARTH_89G_NUM
  INTEGER*2 HLOAD_89G_STRT
  INTEGER*2 HLOAD_89G_NUM
  INTEGER*2 CSKY_89G_STRT
  INTEGER*2 CSKY_89G_NUM
  INTEGER*2 EARTH_166G_STRT
  INTEGER*2 EARTH_166G_NUM
  INTEGER*2 HLOAD_166G_STRT
  INTEGER*2 HLOAD_166G_NUM
  INTEGER*2 CSKY_166G_STRT
  INTEGER*2 CSKY_166G_NUM
  INTEGER*2 EARTH_183G_STRT
INTEGER*2 EARTH_183G_NUM
INTEGER*2 HLOAD_183G_STRT
INTEGER*2 HLOAD_183G_NUM
INTEGER*2 CSKY_183G_STRT
INTEGER*2 CSKY_183G_NUM
END STRUCTURE

STRUCTURE /L1AGMI_S3_MECHANISMS/
  INTEGER*4 SCE_LAST_CMD
  INTEGER*4 SCE_LAST_RESPNS
  INTEGER*4 SCE_CMD_CNT
  INTEGER*2 SMA_RTPRB_SEC
  INTEGER*2 SMA_RATE
  INTEGER*2 SCE_RATE
  INTEGER*2 SCE_CMD_RATE
  INTEGER*2 SMA_CMD_RATE
  INTEGER*2 RESOLVER_POS
  INTEGER*2 MECH_CMD_CNT
  INTEGER*2 SCE_INHIBIT
  INTEGER*2 TACH_PULSE_CNT
  INTEGER*2 LR_ABRT_CNT
  INTEGER*2 RAMP_ABRT_CNT
  INTEGER*2 OVRD_RDA_LR
  INTEGER*2 OVRD_IBS_LR
  INTEGER*2 SCE_PASSPROT
  INTEGER*2 RAMP_INPROGRESS
  INTEGER*2 SMA_SPINNING
  INTEGER*2 LR_RLS_IN_PROG
  INTEGER*2 SCE_A_POWER
  INTEGER*2 SCE_B_POWER
  INTEGER*2 SCE_SELECTION
  INTEGER*2 SMA_RATE_PROB
  INTEGER*2 LR_ENABLED
  INTEGER*2 BILVL_IBS1
  INTEGER*2 BILVL_IBS2
  INTEGER*2 BILVL_IBS3
  INTEGER*2 BILVL_CAL
  INTEGER*2 BILVL_RDALEFT
  INTEGER*2 BILVL_RDARIGHT
  INTEGER*2 BILVL_RDALOWER
END STRUCTURE

STRUCTURE /L1AGMI_S3_IE_TELEMETRY/
INTEGER*4 IE_PASSTHRU_RSP
INTEGER*2 IE_BLANKING_CNT
INTEGER*2 PWR_STAT_LR_PR
INTEGER*2 PWR_STAT_A
INTEGER*2 HTR_STAT_SMA
INTEGER*2 HTR_STAT_RDA
INTEGER*2 HTR_STAT_RS
INTEGER*2 PWR_STAT_LR_RED
INTEGER*2 RS_MST_RLY_STAT
INTEGER*2 RDA_DEPL_STAT_4
INTEGER*2 RDA_DEPL_STAT_3
INTEGER*2 RDA_DEPL_STAT_2
INTEGER*2 RDA_DEPL_STAT_1
INTEGER*2 CLR_STAT_N
INTEGER*2 IBS_LR3_STAT_N
INTEGER*2 MR_LR_LOWR_STAT
INTEGER*2 IBS_LR2_STAT_N
INTEGER*2 MR_LR_LEFT_STAT
INTEGER*2 IBS_LR1_STAT_N
INTEGER*2 MR_LR_RGHT_STAT
INTEGER*2 IE_LATCHUP_CHAN
INTEGER*2 IE_LATCHUP_RETR
INTEGER*2 IE_LATCHUP_SMPS
INTEGER*2 IE_LATCHUP_NUM
INTEGER*2 IE_LATCHUP_FAIL
INTEGER*2 HTR_STAT_HTLD
INTEGER*2 IE_LATCHUP_PAD
END STRUCTURE

STRUCTURE /L1AGMI_S3_IEHSK_TEMP/
  INTEGER*2 IBS_LR1_TEMP
  INTEGER*2 RS_TEMP_1
  INTEGER*2 SCE_A_BD_TEMP
  INTEGER*2 MR_LR_RGHT_TEMP
  INTEGER*2 SMA_BEARING_TMP
  INTEGER*2 PC_BD_TEMP
  INTEGER*2 LVPS_BD_TEMP
  INTEGER*2 SMA_MTR_TEMP
  INTEGER*2 HL_TEMP_2
  INTEGER*2 RDA_TEMP_2
  INTEGER*2 HL_TEMP_1
  INTEGER*2 RS_TEMP_2
  INTEGER*2 SCE_B_BD_TEMP
INTEGER*2 TEMP_CALRES_1
INTEGER*2 TEMP_CALRES_2
INTEGER*2 MR_ICA_TEMP
INTEGER*2 IBS_LR2_TEMP
INTEGER*2 HL_TEMP_7
INTEGER*2 RDA_TEMP_1
INTEGER*2 MR_LR_LEFT_TEMP
INTEGER*2 RDA_TEMP_3
INTEGER*2 ICA_BOX_TEMP_1
INTEGER*2 HL_TEMP_10
INTEGER*2 MR_LR_LOWWR_TEMP
INTEGER*2 SMA_SLPRHTR_TMP
INTEGER*2 HL_TEMP_8
INTEGER*2 TEMP_CALRES_3
INTEGER*2 TEMP_CALRES_4
INTEGER*2 CSR_TEMP1
INTEGER*2 HL_TEMP_12
INTEGER*2 HL_TEMP_13
INTEGER*2 HL_TEMP_14
INTEGER*2 HL_TEMP_11
INTEGER*2 IBS_LR3_TEMP
INTEGER*2 HL_TEMP_9
INTEGER*2 ROT_TEMP_SPARE
INTEGER*2 CSR_TEMP2
INTEGER*2 CE_BD_TEMP
INTEGER*2 ICA_BOX_TEMP_2
INTEGER*2 IE_BD_TEMP
INTEGER*2 CLR_TEMP
INTEGER*2 HL_TEMP_15
INTEGER*2 TEMP_CALRES_5
INTEGER*2 TEMP_CALRES_6
END STRUCTURE

STRUCTURE /L1AGMI_S3_RSHSK_TEMP/
  INTEGER*2 TEMP_10GHZRCVR
  INTEGER*2 TEMP_H166GHZMXR
  INTEGER*2 TEMP_18GHZRCVR
  INTEGER*2 TEMP_V166GHZMXR
  INTEGER*2 TEMP_23GHZRCVR
  INTEGER*2 TEMP_183GHZMXR
  INTEGER*2 TEMP_36GHZRCVR
  INTEGER*2 TEMP_H10GHZ_ND
  INTEGER*2 TEMP_89GHZRCVR
5.1 1AGMI - GMI unpacked packet data

INTEGER*2 TEMP_V10GHZ_ND
INTEGER*2 TEMP_166GHZRCVR
INTEGER*2 TEMP_H18GHZ_ND
INTEGER*2 TEMP_183GHZRCVR
INTEGER*2 TEMP_V18GHZ_ND
INTEGER*2 TEMP_RS_EPC
INTEGER*2 TEMP_H36GHZ_ND
INTEGER*2 TEMP_RS_EDC
INTEGER*2 TEMP_V36GHZ_ND
INTEGER*2 TEMP_FEED
INTEGER*2 TEMP_89GHZ_LO
INTEGER*2 TEMP_HL_TRAY
INTEGER*2 TEMP_166GHZ_LO
INTEGER*2 TEMP_SMASPUNHS
g
INTEGER*2 TEMP_RS_MR1
INTEGER*2 TEMP_H89GHZMXR
INTEGER*2 TEMP_RS_MR2
INTEGER*2 TEMP_V89GHZMXR
INTEGER*2 TEMP_183GHZ_LO
END STRUCTURE

STRUCTURE /L1AGMI_S3_RSHSK_GAIN/
  INTEGER*2 RESERVED2
  INTEGER*2 GAIN_V10GHZ
  INTEGER*2 GAIN_H36GHZ
  INTEGER*2 GAIN_H89GHZ
  INTEGER*2 GAIN_H10GHZ
  INTEGER*2 GAIN_H166GHZ
  INTEGER*2 GAIN_V18GHZ
  INTEGER*2 GAIN_H18GHZ
  INTEGER*2 GAIN_VB183GHZ
  INTEGER*2 GAIN_V23GHZ
  INTEGER*2 GAIN_V36GHZ
  INTEGER*2 GAIN_V89GHZ
  INTEGER*2 GAIN_V166GHZ
  INTEGER*2 GAIN_VA183GHZ
END STRUCTURE

STRUCTURE /L1AGMI_S3_RSHSK_SAMPL_INFO/
  INTEGER*2 SMPOFFST_10GHZ
  INTEGER*2 SMPOFFST_18GHZ
  INTEGER*2 SMPOFFST_23GHZ
  INTEGER*2 SMPOFFST_36GHZ
INTEGER*2 SMPOFFST_89GHZ
INTEGER*2 SMPOFFST_166GHZ
INTEGER*2 SMPOFFST_183GHZ
INTEGER*2 NUMSMPLS_10GHZ
INTEGER*2 NUMSMPLS_18GHZ
INTEGER*2 NUMSMPLS_23GHZ
INTEGER*2 NUMSMPLS_36GHZ
INTEGER*2 NUMSMPLS_89GHZ
INTEGER*2 NUMSMPLS_166GHZ
INTEGER*2 NUMSMPLS_183GHZ

END STRUCTURE

STRUCTURE /L1AGMI_S3_RSHSK_STATUS/
  INTEGER*2 RSST_SCI_ADC_LP
  INTEGER*2 RSST_HSK_ADC_LP
  INTEGER*2 RSST_SAMP_OVLP
  INTEGER*2 RSST_10GHZ_RLY
  INTEGER*2 RSST_18GHZ_RLY
  INTEGER*2 RSST_23GHZ_RLY
  INTEGER*2 RSST_36GHZ_RLY
  INTEGER*2 RSST_89GHZ_RLY
  INTEGER*2 RSST_166GHZ_RLY
  INTEGER*2 RSST_183GHZ_RLY
  INTEGER*2 RSST_INVLD_CMD
  INTEGER*2 RSST_CMD_AFTER
  INTEGER*2 NDIODE_MODE
  INTEGER*2 RSST_NDIODE_ST
  INTEGER*2 NDIODE10GHZSNUM
  INTEGER*2 RESERVED1
  INTEGER*2 RS_CALRES_1
  INTEGER*2 BATC_CALRES_1
  INTEGER*2 RS_CALRES_2
  INTEGER*2 BATC_CALRES_2
  INTEGER*2 RS_EPC_ISENS
  INTEGER*2 RS_EPC_5V
  INTEGER*2 RS_EPC_7V
  INTEGER*2 RS_EPC_POS12V
  INTEGER*2 RS_EPC_NEG12V
  INTEGER*2 RS_EPC_15V

END STRUCTURE

STRUCTURE /L1AGMI_S3_SYNCH_STAMPS2/
  INTEGER*4 TACH_SECS_16
INTEGER*2 TACH_SUBS_16
INTEGER*4 TACH_SECS_17
INTEGER*2 TACH_SUBS_17
INTEGER*4 TACH_SECS_18
INTEGER*2 TACH_SUBS_18
INTEGER*4 TACH_SECS_19
INTEGER*2 TACH_SUBS_19
INTEGER*4 TACH_SECS_20
INTEGER*2 TACH_SUBS_20
INTEGER*4 TACH_SECS_21
INTEGER*2 TACH_SUBS_21
INTEGER*4 TACH_SECS_22
INTEGER*2 TACH_SUBS_22
INTEGER*4 TACH_SECS_23
INTEGER*2 TACH_SUBS_23
INTEGER*4 TACH_SECS_24
INTEGER*2 TACH_SUBS_24
INTEGER*4 TACH_SECS_25
INTEGER*2 TACH_SUBS_25
INTEGER*4 TACH_SECS_26
INTEGER*2 TACH_SUBS_26
INTEGER*4 TACH_SECS_27
INTEGER*2 TACH_SUBS_27
INTEGER*4 TACH_SECS_28
INTEGER*2 TACH_SUBS_28
INTEGER*4 TACH_SECS_29
INTEGER*2 TACH_SUBS_29
INTEGER*4 TACH_SECS_30
INTEGER*2 TACH_SUBS_30
INTEGER*4 TACH_SECS_31
INTEGER*2 TACH_SUBS_31
INTEGER*4 SCAN_COMPL_SECS
INTEGER*2 SCAN_COMPL_SUBS
END STRUCTURE

STRUCTURE /L1AGMI_S3_SYNCH_STAMPS/
  INTEGER*4 IDX_PULSE_SECS
  INTEGER*2 IDX_PULSE_SUBS
  INTEGER*4 TACH_SECS_00
  INTEGER*2 TACH_SUBS_00
  INTEGER*4 TACH_SECS_01
  INTEGER*2 TACH_SUBS_01
  INTEGER*4 TACH_SECS_02
INTEGER*2 TACH_SUBS_02
INTEGER*4 TACH_SECS_03
INTEGER*2 TACH_SUBS_03
INTEGER*4 TACH_SECS_04
INTEGER*2 TACH_SUBS_04
INTEGER*4 TACH_SECS_05
INTEGER*2 TACH_SUBS_05
INTEGER*4 TACH_SECS_06
INTEGER*2 TACH_SUBS_06
INTEGER*4 TACH_SECS_07
INTEGER*2 TACH_SUBS_07
INTEGER*4 TACH_SECS_08
INTEGER*2 TACH_SUBS_08
INTEGER*4 TACH_SECS_09
INTEGER*2 TACH_SUBS_09
INTEGER*4 TACH_SECS_10
INTEGER*2 TACH_SUBS_10
INTEGER*4 TACH_SECS_11
INTEGER*2 TACH_SUBS_11
INTEGER*4 TACH_SECS_12
INTEGER*2 TACH_SUBS_12
INTEGER*4 TACH_SECS_13
INTEGER*2 TACH_SUBS_13
INTEGER*4 TACH_SECS_14
INTEGER*2 TACH_SUBS_14
INTEGER*4 TACH_SECS_15
INTEGER*2 TACH_SUBS_15

END STRUCTURE

STRUCTURE /L1AGMI_S3_RS_INFO/
    INTEGER*2 RS_POWERED
    INTEGER*2 RS_ENABLED
    INTEGER*2 RS_MST_RLY
    INTEGER*2 RS_10GHZ_RLY
    INTEGER*2 RS_18GHZ_RLY
    INTEGER*2 RS_23GHZ_RLY
    INTEGER*2 RS_36GHZ_RLY
    INTEGER*2 RS_89GHZ_RLY
    INTEGER*2 RS_166GHZ_RLY
    INTEGER*2 RS_183GHZ_RLY
    INTEGER*2 RS_DQ_MISSING
    INTEGER*2 RS_DQ_EXTRAS
    INTEGER*2 RS_DQ_DUPES
STRUCTURE /L1AGMI_S3_SENSOR_INFO/
  INTEGER*4 KEEP_ALIVE_CNT
  INTEGER*2 FPGA_RST_REASON
  INTEGER*2 CRASH_REASON
  INTEGER*2 VERSION_MIN
  INTEGER*2 VERSION_MAJ
  INTEGER*2 FPGA_MODE
  INTEGER*2 ERR_HDL_FAILURE
  INTEGER*2 RESET_REASON
  INTEGER*2 BOOT_BANK
  INTEGER*2 CURRENT_BANK
  INTEGER*2 EDAC_ENABLE
  INTEGER*2 WDOG_ENABLE
END STRUCTURE
INTEGER*2 SC_1HZ_REF
INTEGER*2 SCE_FORCE_SEL
INTEGER*2 RS_1MHZ_REF
INTEGER*2 RS_SCAN_START
INTEGER*2 FPGA_IE_RX_EN
INTEGER*2 FPGA_RS_RX_EN
INTEGER*2 FPGA_SCE_RX_EN
INTEGER*2 EEPROM_BUSY
INTEGER*2 RS_TLM_PROG
INTEGER*2 SCE_A_ACTIVE
INTEGER*2 SCE_A_RLY
INTEGER*2 SCE_B_ACTIVE
INTEGER*2 SCE_B_RLY
INTEGER*2 IE_TLM_PROG
INTEGER*2 SCE_RSP_PROG
INTEGER*2 RS_CLK_ERR
INTEGER*2 RS_PKT_ERR
INTEGER*2 RS_TLM_ERR
INTEGER*2 SCE_RSP_RDY
INTEGER*2 IE_PKT_ERR
INTEGER*2 IE_CMD_ERR
INTEGER*2 IE_RSP_ERR
INTEGER*2 IE_TLM_ERR
INTEGER*2 FPGA_ACCSS_ERR
END STRUCTURE

STRUCTURE /L1AGMI_S3_GSDR_TIME/
INTEGER*4 G_TC_PULSE_SECS
INTEGER*2 G_TC_PULSE_SUBS
INTEGER*4 G_TCU_SECS
INTEGER*4 G_TCU_SUBS
INTEGER*4 G_TCF_SC_SECS
INTEGER*4 G_TCF_SC_SUBSEC
INTEGER*4 G_TCF_SECS
INTEGER*4 G_TCF_SUBSECS
INTEGER*2 G_TCF_SIGN
INTEGER*2 G_TCF_LEAP
INTEGER*4 GPS_TCU_SECS
INTEGER*4 GPS_TCU_SUBS
END STRUCTURE

STRUCTURE /PRIMARYHEADER/
BYTE version
5.1 1AGMI - GMI unpacked packet data

BYTE type
BYTE secHeaderFlag
INTEGER*2 APID
BYTE sequenceFlag
INTEGER*2 packetSequenceCount
INTEGER*2 packetLength
END STRUCTURE

STRUCTURE /L1AGMI_S3_GMI_TEMPERATURES/
    REAL*4 timeOffset
    INTEGER*2 apid
    INTEGER*2 SMA_PT_TEMP
    INTEGER*2 ICA_PT_TEMP
    INTEGER*2 RS_PT_TEMP
    INTEGER*2 STAT_PT_TEMP
    INTEGER*2 MR_PT_TEMP
END STRUCTURE

STRUCTURE /L1AGMI_S3_TORQUE_BAR/
    REAL*4 timeOffset
    INTEGER*2 Vx
    INTEGER*2 Vy
    INTEGER*2 Vz
END STRUCTURE

STRUCTURE /L1AGMI_S3_TAM2/
    REAL*4 timeOffset
    INTEGER*2 Vx
    INTEGER*2 Vy
    INTEGER*2 Vz
END STRUCTURE

STRUCTURE /L1AGMI_S3_TAM1/
    REAL*4 timeOffset
    INTEGER*2 Vx
    INTEGER*2 Vy
    INTEGER*2 Vz
END STRUCTURE

STRUCTURE /L1AGMI_S3/
    RECORD /SCANTIME/ ScanTime
    CHARACTER ephemerisUsed(10)
    REAL*4 Latitude
REAL*4 Longitude
RECORD /L1AGMI_S3_TAM1/ TAM1
RECORD /L1AGMI_S3_TAM2/ TAM2
RECORD /L1AGMI_S3_TORQUE_BAR/ TORQUE_BAR
RECORD /L1AGMI_S3_GMI_TEMPERATURES/ GMI_TEMPERATURES
RECORD /PRIMARYHEADER/ primaryHeader
INTEGER*4 instrTimeSeconds
INTEGER*2 instrTimeSubseconds
BYTE numPacketSegments
BYTE spare
INTEGER*2 RDRversion
RECORD /L1AGMI_S3_GSDR_TIME/ GSDR_TIME
RECORD /L1AGMI_S3_SENSOR_INFO/ SENSOR_INFO
RECORD /L1AGMI_S3_RS_INFO/ RS_INFO
RECORD /L1AGMI_S3_SYNCH_STAMPS/ SYNCH_STAMPS
RECORD /L1AGMI_S3_SYNCH_STAMPS2/ SYNCH_STAMPS2
RECORD /L1AGMI_S3_RSHSK_STATUS/ RSHSK_STATUS
RECORD /L1AGMI_S3_RSHSK_SAMPL_INFO/ RSHSK_SAMPL_INFO
RECORD /L1AGMI_S3_RSHSK_GAIN/ RSHSK_GAIN
RECORD /L1AGMI_S3_RSHSK_TEMP/ RSHSK_TEMP
RECORD /L1AGMI_S3_IEHSK_TEMP/ IEHSK_TEMP
RECORD /L1AGMI_S3_IE_TELEMETRY/ IE_TELEMETRY
RECORD /L1AGMI_S3_MECHANISMS/ MECHANISMS
RECORD /L1AGMI_S3_SMPL_INFO/ SMPL_INFO
END STRUCTURE

STRUCTURE /L1AGMI_S2_SUNDATA/
   REAL*4 solarBetaAngle
   REAL*4 phaseFromOrbitMidnight
   REAL*4 sunEarthSeparation
   REAL*4 earthAngularRadius
   REAL*4 phaseOfEclipseExit
   REAL*4 orbitRate
   REAL*4 timeSinceEclipseEntry
   REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1AGMI_S2_SCANSTATUS/
   BYTE dataQuality
   BYTE missing
   BYTE modeStatus
   INTEGER*2 geoError
   INTEGER*2 geoWarning
5.1 1AGMI - GMI unpacked packet data

INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /L1AGMI_S2/

RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(221)
REAL*4 Longitude(221)
RECORD /L1AGMI_S2SCANSTATUS/ scanStatus
CHARACTER ephemerisUsed(10)
RECORD /NAVIGATION/ navigation
RECORD /L1AGMI_S2_SUNDATA/ sunData
REAL*4 incidenceAngle(221)
REAL*4 satAzimuthAngle(221)
REAL*4 solarZenAngle(221)
REAL*4 solarAzimuthAngle(221)
REAL*4 sunGlintAngle(221)
REAL*4 moonVectorInstFrame(3)
INTEGER*2 earthView(4,221)
INTEGER*2 hotLoad(4,221)
INTEGER*2 coldSky(4,221)

END STRUCTURE

STRUCTURE /L1AGMI_S1_SUNDATA/

REAL*4 solarBetaAngle
REAL*4 phaseFromOrbitMidnight
REAL*4 sunEarthSeparation
REAL*4 earthAngularRadius
REAL*4 phaseOfEclipseExit
REAL*4 orbitRate
REAL*4 timeSinceEclipseEntry
REAL*4 sunVectorInBodyFrame(3)

END STRUCTURE

STRUCTURE /NAVIGATION/

REAL*4 scPos(3)
REAL*4 scVel(3)
REAL*4 scLat
REAL*4 scLon
REAL*4 scAlt
REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1AGMI_S1_SCANSTATUS/
BYTE dataQuality
BYTE missing
BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
INTEGER*2 Year
BYTE Month
BYTE DayOfMonth
BYTE Hour
BYTE Minute
BYTE Second
INTEGER*2 MilliSecond
INTEGER*2 DayOfYear
REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1AGMI_S1/
RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(221)
REAL*4 Longitude(221)
5.2 1ATMI - TMI unpacked packet data

1ATMI contains unpacked packet data from TMI science data from the TMI passive microwave instrument flown on the TRMM satellite. There are 4 swaths. Swath S1 has 10V 10H; Swath S2 has 19V, 19H, 21V, 37V, 37H; Swath S3 has 85V, 85H; Swath S4 has Housekeeping.

The S1 channels are:
10.7 GHz vertically-polarized
10.7 GHz horizontally-polarized

The S2 channels are:
18.7 GHz vertically-polarized
18.7 GHz horizontally-polarized
23.8 GHz vertically-polarized
36.5 GHz vertically-polarized
36.5 GHz horizontally-polarized

The S3 channels are:
85.0 GHz vertically-polarized
85.0 GHz horizontally-polarized

S4 has TMI housekeeping.

Earth observations are taken during a segment of the rotation when TMI is looking in the +x direction of the TRMM satellite. Since the spacecraft turns around every few weeks, +x may be forward or aft. We define the spacecraft axis v, used in the definition of the variable SCorientation, at the center of this segment and the same as the +x direction.

Before Aug 7, 2001 31.6rpm * 1min/60s * 5490s/orbit = 2891 scans / orbit.

After Aug 24, 2001 31.6rpm * 1min/60s * 5550s/orbit = 2923 scans / orbit.

RELATION BETWEEN THE SWATHS: Swath S2 has the same number of scans and the same number of pixels as Swath S1. Swath S3 has the same number of scans and twice as many pixels as Swath S1. Each S1 scan contains 2 channels sampled 104 times along the scan. Each S2 scan contains 5 channels sampled 104 times along the scan. Each S3 scan contains 2 channels sampled 208 times along the scan.

Dimension definitions:
nsoparm  3  Number of swath offset parameters: cone angle, start angle, first
pixel time.

nswath  3  Number of swaths, not counting housekeeping swath.

VH  2  Number of polarizations.

nscan1  var  Typical number of Swath S1 scans in the granule.

nchannel1  2  Number of Swath S1 channels (10V).

npixelev1  104  Number of earth view pixels in one scan.

npixelht1  8  Number of hot load pixels in one scan.

npixels1  8  Number of cold sky pixels in one scan.

nscan2  var  Typical number of Swath S2 scans in the granule.

nchannel2  5  Number of Swath S2 channels (19V 19H 21V 37V 37H).

npixelev2  104  Number of earth view pixels in one scan.

npixelht2  8  Number of hot load pixels in one scan.

npixels2  8  Number of cold sky pixels in one scan.

nscan3  var  Typical number of Swath S3 scans in the granule.

nchannel3  2  Number of Swath S3 channels (85V 85H).

npixelev3  208  Number of earth view pixels in one scan.

npixelht3  16  Number of hot load pixels in one scan.

npixels3  16  Number of cold sky pixels in one scan.

nscan4  var  Typical number of Swath S4 scans in the granule.

nchannelall  9  Number of all channels.

dim2  2  Number.

dim3  3  Number.

dim4  4  Number.

dim5  5  Number.

dim6  6  Number.

dim7  7  Number.

dim8  8  Number.

dim9  9  Number.

dim10  10  Number.

dim11  11  Number.

dim12  12  Number.

TMIxyz  3  x, y, z components in TMI instrument coordinate system.

Figure 48 through Figure 72 show the structure of this product. The text below describes
the contents of objects in the structure, the C Structure Header File and the Fortran
Structure Header File.
Figure 48: Data Format Structure for 1ATMI, TMI unpacked packet data
5.2 1ATMI - TMI unpacked packet data

Figure 49: Data Format Structure for 1ATMI, S1
Figure 50: Data Format Structure for 1ATMI, S2
5.2 1ATMI - TMI unpacked packet data

Figure 51: Data Format Structure for 1ATMI, S3
Figure 52: Data Format Structure for 1ATMI, S4

Figure 53: Data Format Structure for 1ATMI, tmi1aHeader

Figure 54: Data Format Structure for 1ATMI, S1, ScanTime
5.2 1ATMI - TMI unpacked packet data

![Diagram of scanStatus structure]

- dataQuality 1 byte Array: nscan1
- missing 1 byte Array: nscan1
- modeStatus 1 byte Array: nscan1
- geoError 2 bytes Array: nscan1
- geoWarning 2 bytes Array: nscan1
- SCorientation 2 bytes Array: nscan1
- pointingStatus 2 bytes Array: nscan1
- acsModeMidScan 1 byte Array: nscan1
- targetSelectionMidScan 1 byte Array: nscan1
- tmIsStatus 1 byte Array: nscan1
- FractionalGranuleNumber 8 bytes Array: nscan1
- attDetermSource 2 bytes Array: nscan1
- TRMMcontMode 1 byte Array: nscan1
- TRMMyawUpdateS 1 byte Array: nscan1
- TRMMqac 1 byte Array: nscan1

Figure 55: Data Format Structure for 1ATMI, S1, scanStatus

![Diagram of primaryHeader structure]

- version 1 byte Array: nscan1
- type 1 byte Array: nscan1
- secHeaderFlag 1 byte Array: nscan1
- APID 2 bytes Array: nscan1
- sequenceFlag 1 byte Array: nscan1
- packetSequenceCount 2 bytes Array: nscan1
- packetLength 2 bytes Array: nscan1

Figure 56: Data Format Structure for 1ATMI, S1, primaryHeader
Figure 57: Data Format Structure for 1ATMI, S1, navigation

Figure 58: Data Format Structure for 1ATMI, S1, sunData
5.2 1ATMI - TMI unpacked packet data

Figure 59: Data Format Structure for 1ATMI, S2, ScanTime

Figure 60: Data Format Structure for 1ATMI, S2, scanStatus
Figure 61: Data Format Structure for 1ATMI, S2, primaryHeader
### FileHeader (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

### InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

### AlgorithmRuntimeInfo (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

### NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

### FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.
5 STANDARD GPM PRODUCTS

Figure 63: Data Format Structure for 1ATMI, S2, sunData

Figure 64: Data Format Structure for 1ATMI, S3, ScanTime
5.2 1ATMI - TMI unpacked packet data

Figure 65: Data Format Structure for 1ATMI, S3, scanStatus

Figure 66: Data Format Structure for 1ATMI, S3, primaryHeader
Figure 67: Data Format Structure for 1ATMI, S3, navigation

Figure 68: Data Format Structure for 1ATMI, S3, sunData
5.2 1ATMI - TMI unpacked packet data

Figure 69: Data Format Structure for 1ATMI, S4, ScanTime

Figure 70: Data Format Structure for 1ATMI, S4, primaryHeader
Figure 71: Data Format Structure for 1ATMI, TMIHK PACKET

continued on next figure
5.2 1ATMI - TMI unpacked packet data

Figure 72: Data Format Structure for 1ATMI, S4, TMIHKPACKET

**tmi1aHeader** (Group)

**swathOffsets** (4-byte float, array: nsoparm x nswath): Angle and timing offsets for each swath. Swath index (4) is 10V 10H 37 85 GHz. Parameter index (3) is cone angle (degrees), start angle (degrees), first pixel time (seconds).

**chnl10AngleDiff** (4-byte float, array size: 1): Cone angle offsets for 10V/H channels used for calculation of incidence angles. Negative offset for 10V channel, positive offset for 10H channel. Values range from 0.0 to 1.0 degrees. Special values are defined as:
-9999.9 Missing value

**S1** (Swath)

**S1_SwathHeader** (Metadata): SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
ScanTime (Group in S1)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
  -9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
  -99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
  -99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
  -99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
  -99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
  -99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
  -9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
  -9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
  -9999.9 Missing value

Latitude (4-byte float, array size: npixelev1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
  -9999.9 Missing value

Longitude (4-byte float, array size: npixelev1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod.
Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S1)

**dataQuality** (1-byte char, array size: nscan1):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not normal</td>
</tr>
<tr>
<td>7</td>
<td>QAC errors associated with this scan</td>
</tr>
</tbody>
</table>

**missing** (1-byte char, array size: nscan1):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte char, array size: nscan1):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation is not 0 or 180</td>
</tr>
</tbody>
</table>
geoError (2-byte integer, array size: nscan1):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i). Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoWarning (2-byte integer, array size: nscan1):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
Bit Meaning if bit = 1
0 Ephemeris Gap Interpolated
1 Attitude Gap Interpolated
2 Attitude jump/discontinuity
3 Attitude out of range
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

SCrientation (2-byte integer, array size: nscan1):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

Value Meaning
0 +X forward (yaw 0)
90 -Y forward (yaw 90)
180 -X forward (yaw 180)
-8002 Yaw turn in progress
-8003 Deep Space Calibration in progress
-8004 Non-nominal pointing other than above
-9999 Missing

pointingStatus (2-byte integer, array size: nscan1):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

Value Meaning
0 Nominal ACS mode (4) for mission science
-8000 Non-nominal ACS mode
**acsModeMidScan** (1-byte integer, array size: nscan1):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: nscan1):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**tmiIsStatus** (1-byte char, array size: nscan1):
Status of the instrument from Housekeeping packets. Bit 0 is the most significant bit (I.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^{*(8-i)} - 1$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>01</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>02</td>
<td>Spare command 1 Status</td>
</tr>
<tr>
<td>03</td>
<td>Spare command 2 Status</td>
</tr>
<tr>
<td>04</td>
<td>1 Hz Clock Select (1=A, 0=B)</td>
</tr>
<tr>
<td>05</td>
<td>Spare</td>
</tr>
<tr>
<td>06</td>
<td>Spare Command 4 Status</td>
</tr>
<tr>
<td>07</td>
<td>Spare Command 5 Status</td>
</tr>
</tbody>
</table>

**FractionalGranuleNumber** (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**attDetermSource** (2-byte integer, array size: nscan1):

Attitude determination source.
A flag explaining how the attitude value was calculated.
Improved estimates make use of ground processing of PR
science-instrument-measured roll values, Gyroscope data,
and Sun Sensor 1 data. Earlier products (TRMM V7 and before)
used the onboard attitudes with various corrections.
Values were determined for each granule based on the data
available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430 and higher</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
<tr>
<td>421</td>
<td>Reduced accuracy, PR roll data not available</td>
</tr>
<tr>
<td></td>
<td>(affecting roll/yaw estimates)</td>
</tr>
<tr>
<td>413</td>
<td>Reduced accuracy, sun data not available (affecting pitch)</td>
</tr>
<tr>
<td>411</td>
<td>Reduced accuracy, PR roll and sun sensor not available</td>
</tr>
<tr>
<td>300-399</td>
<td>Reduced accuracy due to various special conditions</td>
</tr>
<tr>
<td>200-299</td>
<td>Fallback to using the onboard attitude estimates</td>
</tr>
<tr>
<td></td>
<td>with TRMM V7 corrections</td>
</tr>
<tr>
<td>-91</td>
<td>Spacecraft in safefold mode, no science data</td>
</tr>
<tr>
<td>-99</td>
<td>No data due to telemetry data gap</td>
</tr>
</tbody>
</table>

**TRMMcontMode** (1-byte integer, array size: nscan1):

The Contingency Mode Flag from telemetry indicates
alternate attitude control of the spacecraft.
The nominal at-launch Attitude Control System (ACS)
for TRMM used Earth horizon sensors for pitch and
roll control, and the yaw was updated twice each orbit
using the Sun Sensors and propagated using gyro data.
However, due to possible problems identified with
the Earth Sensor Assembly (ESA) lifetime on-orbit,
a contingency ACS mode was developed late in the
development cycle. This mode used the Sun Sensors,
magnetometers, and gyroscope data. It proved very
valuable when the horizon sensors had problems with
TRMM moving to the higher operating altitude (from
350 to 402.5 km) to extend the mission lifetime.
Thus the contingency mode was used throughout the
post-boost period. It was also tested early in the mission on 1998-01-13.

Value Meaning
0 Nominal control of spacecraft used in the pre-boost period
1 Contingency mode control used in the post-boost period
-99 Missing

TRMMyawUpdateS (1-byte integer, array size: nscan1):

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

Value Meaning
0 Inaccurate
1 Indeterminate
2 Accurate
-99 Missing

TRMMqac (1-byte integer, array size: nscan1):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

ephemerisUsed (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
255 Missing value

primaryHeader (Group in S1)

version (1-byte integer, array size: nscan1):

type (1-byte integer, array size: nscan1):
5.2 1ATMI - TMI unpacked packet data

secHeaderFlag (1-byte integer, array size: nscan1):

APID (2-byte integer, array size: nscan1):

sequenceFlag (1-byte integer, array size: nscan1):

packetSequenceCount (2-byte integer, array size: nscan1):

packetLength (2-byte unsigned integer, array size: nscan1):

navigation (Group in S1)

scPos (4-byte float, array size: XYZ x nscan1):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
   -9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan1):
The velocity vector (m s\(^{-1}\)) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
   -9999.9 Missing value

scLat (4-byte float, array size: nscan1):
The geodetic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
   -9999.9 Missing value

scLon (4-byte float, array size: nscan1):
The geodetic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9 Missing value

scAlt (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
   -9999.9 Missing value

dprAlt (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
   -9999.9 Missing value
**scAttRollGeoc** (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value
greenHourAng (4-byte float, array size: nscan1):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coor-
dinates. Values range from 0 to 360 degrees. Special values are defined as:
  -9999.9  Missing value

timeMidScan (8-byte float, array size: nscan1):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC,6 Jan 1980.
timeMidScan is used as the reference time for the scPos and scVel values. Values range
from 0 to 1000000000 s. Special values are defined as:
  -9999.9  Missing value

timeMidScanOffset (8-byte float, array size: nscan1):
Offset from the secondary header packet time to the timeMidScan. Values range from 0
to 100 s. Special values are defined as:
  -9999.9  Missing value

sunData (Group in S1)
solarBetaAngle (4-byte float, array size: nscan1):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given
by the cross product of the spacecraft position and velocity vectors. Values range from
-89.0 to 89.0 degrees. Special values are defined as:
  -9999.9  Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan1):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of
the Earth center from the spacecraft and positive toward the spacecraft velocity direction
so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees
occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local
orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator.
Values range from -180.0 to 180.0 degrees. Special values are defined as:
  -9999.9  Missing value

sunEarthSeparation (4-byte float, array size: nscan1):
The separation angle between the Sun and Earth directions from the spacecraft. Values
range from 0 to 180.0 degrees. Special values are defined as:
  -9999.9  Missing value

earthAngularRadius (4-byte float, array size: nscan1):
The angle between the center of the Earth and the horizon edge. The sun is above
the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius.
Values range from 69.0 to 80.0 degrees. Special values are defined as:
  -9999.9  Missing value

phaseOfEclipseExit (4-byte float, array size: nscan1):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow,
based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:

-9999.9 Missing value

**orbitRate** (4-byte float, array size: nscan1):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:

-9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan1):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:

-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan1):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:

-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: nchannel1 x npixelev1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:

-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npixelev1 x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: npixelev1 x nscan1):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: npixelev1 x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npixelev1 x nscan1):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like)
sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**moonVectorInstFrame** (4-byte float, array size: TMIxyz x nscan1):
The x, y, z components of the moon vector in the TMI instrument coordinate system. Values are in counts. Special values are defined as:
-9999.9 Missing value

**earthView** (2-byte unsigned integer, array size: nchannel1 x npixelev1 x nscan1):
Earth view counts.

Special values are defined as:
0 Missing value

**hotLoad** (2-byte unsigned integer, array size: nchannel1 x npixelht1 x nscan1):
Hot load counts.

Special values are defined as:
0 Missing value

**coldSky** (2-byte unsigned integer, array size: nchannel1 x npixelcs1 x nscan1):
Cold sky counts.

Special values are defined as:
0 Missing value

**S2** (Swath)

**S2_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S2)
A UTC time associated with the scan.
**Year** (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan2):
Thousands of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npixelev2 x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixelev2 x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
scanStatus (Group in S2)

**dataQuality** (1-byte char, array size: nscan2):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not normal</td>
</tr>
<tr>
<td>7</td>
<td>QAC errors associated with this scan</td>
</tr>
</tbody>
</table>

**missing** (1-byte char, array size: nscan2):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte char, array size: nscan2):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorotation is not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine tmiIsStatus</td>
</tr>
</tbody>
</table>
geoError (2-byte integer, array size: nscan2):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0 Latitude limit exceeded for viewed pixel locations
1 Negative scan time, invalid input
2 Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

geoWarning (2-byte integer, array size: nscan2):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$):

Bit Meaning if bit = 1
5.2 1ATMI - TMI unpacked packet data

0  Ephemeris Gap Interpolated
1  Attitude Gap Interpolated
2  Attitude jump/discontinuity
3  Attitude out of range
4  Anomalous Time Step
5  GHA not calculated due to error
6  SunData (Group) not calculated due to error
7  Failure to calculate Sun in inertial coordinates
8  Fallback to GES ephemeris
9  Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

SCorientation (2-byte integer, array size: nscan2):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8002</td>
<td>Yaw turn in progress</td>
</tr>
<tr>
<td>-8003</td>
<td>Deep Space Calibration in progress</td>
</tr>
<tr>
<td>-8004</td>
<td>Non-nominal pointing other than above</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan2):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal ACS mode (4) for mission science</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal ACS mode</td>
</tr>
</tbody>
</table>
acsModeMidScan (1-byte integer, array size: nscan2):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan2):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

tmiIsStatus (1-byte char, array size: nscan2):
Status of the instrument from Housekeeping packets. Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(8-i) - 1).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>01</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>02</td>
<td>Spare command 1 Status</td>
</tr>
<tr>
<td>03</td>
<td>Spare command 2 Status</td>
</tr>
<tr>
<td>04</td>
<td>1 Hz Clock Select (1=A, 0=B)</td>
</tr>
<tr>
<td>05</td>
<td>Spare</td>
</tr>
<tr>
<td>06</td>
<td>Spare Command 4 Status</td>
</tr>
<tr>
<td>07</td>
<td>Spare Command 5 Status</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan2):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule.
5.2 1ATMI - TMI unpacked packet data

Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

attDetermSource (2-byte integer, array size: nscan2):

Attitude determination source.
A flag explaining how the attitude value was calculated.
Improved estimates make use of ground processing of PR
science-instrument-measured roll values, Gyroscope data,
and Sun Sensor 1 data. Earlier products (TRMM V7 and before)
used the onboard attitudes with various corrections.
Values were determined for each granule based on the data
available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430 and higher</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
<tr>
<td>421</td>
<td>Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)</td>
</tr>
<tr>
<td>413</td>
<td>Reduced accuracy, sun data not available (affecting pitch)</td>
</tr>
<tr>
<td>411</td>
<td>Reduced accuracy, PR roll and sun sensor not available</td>
</tr>
<tr>
<td>300-399</td>
<td>Reduced accuracy due to various special conditions</td>
</tr>
<tr>
<td>200-299</td>
<td>Fallback to using the onboard attitude estimates with TRMM V7 corrections</td>
</tr>
<tr>
<td>-91</td>
<td>Spacecraft in safehold mode, no science data</td>
</tr>
<tr>
<td>-99</td>
<td>No data due to telemetry data gap</td>
</tr>
</tbody>
</table>

TRMMcontMode (1-byte integer, array size: nscan2):

The Contingency Mode Flag from telemetry indicates
alternate attitude control of the spacecraft.
The nominal at-launch Attitude Control System (ACS)
for TRMM used Earth horizon sensors for pitch and
roll control, and the yaw was updated twice each orbit
using the Sun Sensors and propagated using gyro data.
However, due to possible problems identified with
the Earth Sensor Assembly (ESA) lifetime on-orbit,
a contingency ACS mode was developed late in the
development cycle. This mode used the Sun Sensors,
magnetometers, and gyroscope data. It proved very
valuable when the horizon sensors had problems with
TRMM moving to the higher operating altitude (from
350 to 402.5 km) to extend the mission lifetime.
Thus the contingency mode was used throughout the
post-boost period. It was also tested early in the mission on 1998-01-13.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal control of spacecraft used in the pre-boost period</td>
</tr>
<tr>
<td>1</td>
<td>Contingency mode control used in the post-boost period</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMYawUpdateS** (1-byte integer, array size: nscan2):

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMqac** (1-byte integer, array size: nscan2):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

**ephemerisUsed** (1-byte char, array size: dim10 x nscan2):
The ephemeris source used to geolocate the swath. Special values are defined as:

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>255</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

**primaryHeader** (Group in S2)

**version** (1-byte integer, array size: nscan2):

**type** (1-byte integer, array size: nscan2):
secHeaderFlag (1-byte integer, array size: nscan2):

APID (2-byte integer, array size: nscan2):

sequenceFlag (1-byte integer, array size: nscan2):

packetSequenceCount (2-byte integer, array size: nscan2):

packetLength (2-byte unsigned integer, array size: nscan2):

**navigation** (Group in S2)

scPos (4-byte float, array size: XYZ x nscan2):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan2):
The velocity vector ($m s^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan2):
The geodetic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan2):
The geodetic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan2):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan2):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value
**scAttRollGeoc** (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value
greenHourAng (4-byte float, array size: nscan2):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan2):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan2):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

donData (Group in S2)

solarBetaAngle (4-byte float, array size: nscan2):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan2):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan2):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan2):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan2):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow,
based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**orbitRate** (4-byte float, array size: nscan2):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan2):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan2):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npixelx2 x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npixelx2 x nscan2):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: npixelx2 x nscan2):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: npixelx2 x nscan2):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npixelx2 x nscan2):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like)
sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**moonVectorInstFrame** (4-byte float, array size: TMIxyz x nscan2):
The x, y, z components of the moon vector in the TMI instrument coordinate system.
Values are in counts. Special values are defined as:
-9999.9 Missing value

**earthView** (2-byte unsigned integer, array size: nchannel2 x npixelev2 x nscan2):
Earth view counts.

Special values are defined as:
0 Missing value

**hotLoad** (2-byte unsigned integer, array size: nchannel2 x npixelht2 x nscan2):
Hot load counts.

Special values are defined as:
0 Missing value

**coldSky** (2-byte unsigned integer, array size: nchannel2 x npixelcs2 x nscan2):
Cold sky counts.

Special values are defined as:
0 Missing value

**S3** (Swath)

**S3_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S3)
A UTC time associated with the scan.
Year (2-byte integer, array size: nscan3):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan3):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan3):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan3):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan3):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan3):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan3):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan3):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan3):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixelev3 x nscan3):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixelev3 x nscan3):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
scanStatus (Group in S3)

dataQuality (1-byte char, array size: nscan3):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not normal</td>
</tr>
<tr>
<td>7</td>
<td>QAC errors associated with this scan</td>
</tr>
</tbody>
</table>

missing (1-byte char, array size: nscan3):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

modeStatus (1-byte char, array size: nscan3):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation is not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine tmIsStatus</td>
</tr>
</tbody>
</table>
5 Spare (always 0)  
6 Spare (always 0)  
7 Spare (always 0)  

geoError (2-byte integer, array size: nscan3):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.  
A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken  
down into the bit flags below indicates the specified reason, where bit 0 is the least  
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the  
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of  
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,  
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to  
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoWarning (2-byte integer, array size: nscan3):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in  
dataQuality. Warnings indicate unusual conditions. These conditions do not indicate  
bad geolocation but are flagged as a warning that further review of the data may be  
useful. A zero integer value indicates usual geolocation. A non-zero value broken  
down into the following bit flags indicates the following, where bit 0 is the least  
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
0  Ephemeris Gap Interpolated
1  Attitude Gap Interpolated
2  Attitude jump/discontinuity
3  Attitude out of range
4  Anomalous Time Step
5  GHA not calculated due to error
6  SunData (Group) not calculated due to error
7  Failure to calculate Sun in inertial coordinates
8  Fallback to GES ephemeris
9  Fallback to GEONS ephemeris
10  Fallback to PVT ephemeris
11  Fallback to OBP ephemeris
12  Spare (always 0)
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

**SCorientation** (2-byte integer, array size: nscan3):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8002</td>
<td>Yaw turn in progress</td>
</tr>
<tr>
<td>-8003</td>
<td>Deep Space Calibration in progress</td>
</tr>
<tr>
<td>-8004</td>
<td>Non-nominal pointing other than above</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan3):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal ACS mode (4) for mission science</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal ACS mode</td>
</tr>
</tbody>
</table>
acsModeMidScan (1-byte integer, array size: nscan3):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan3):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

tmiIsStatus (1-byte char, array size: nscan3):
Status of the instrument from Housekeeping packets. Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is \(2^{8-i}-1\)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>01</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>02</td>
<td>Spare command 1 Status</td>
</tr>
<tr>
<td>03</td>
<td>Spare command 2 Status</td>
</tr>
<tr>
<td>04</td>
<td>1 Hz Clock Select (1=A, 0=B)</td>
</tr>
<tr>
<td>05</td>
<td>Spare</td>
</tr>
<tr>
<td>06</td>
<td>Spare Command 4 Status</td>
</tr>
<tr>
<td>07</td>
<td>Spare Command 5 Status</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan3):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule.
5.2 1ATMI - TMI unpacked packet data

Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**attDetermSource** (2-byte integer, array size: nscan3):

Attitude determination source.
A flag explaining how the attitude value was calculated.
Improved estimates make use of ground processing of PR
science-instrument-measured roll values, Gyroscope data,
and Sun Sensor 1 data. Earlier products (TRMM V7 and before)
used the onboard attitudes with various corrections.
Values were determined for each granule based on the data
available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430 and higher</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
<tr>
<td>421</td>
<td>Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)</td>
</tr>
<tr>
<td>413</td>
<td>Reduced accuracy, sun data not available (affecting pitch)</td>
</tr>
<tr>
<td>411</td>
<td>Reduced accuracy, PR roll and sun sensor not available</td>
</tr>
<tr>
<td>300-399</td>
<td>Reduced accuracy due to various special conditions</td>
</tr>
<tr>
<td>200-299</td>
<td>Fallback to using the onboard attitude estimates with TRMM V7 corrections</td>
</tr>
<tr>
<td>-91</td>
<td>Spacecraft in safehold mode, no science data</td>
</tr>
<tr>
<td>-99</td>
<td>No data due to telemetry data gap</td>
</tr>
</tbody>
</table>

**TRMMcontMode** (1-byte integer, array size: nscan3):

The Contingency Mode Flag from telemetry indicates
alternate attitude control of the spacecraft.
The nominal at-launch Attitude Control System (ACS)
for TRMM used Earth horizon sensors for pitch and
roll control, and the yaw was updated twice each orbit
using the Sun Sensors and propagated using gyro data.
However, due to possible problems identified with
the Earth Sensor Assembly (ESA) lifetime on-orbit,
a contingency ACS mode was developed late in the
development cycle. This mode used the Sun Sensors,
magnetometers, and gyroscope data. It proved very
valuable when the horizon sensors had problems with
TRMM moving to the higher operating altitude (from
350 to 402.5 km) to extend the mission lifetime.
Thus the contingency mode was used throughout the
post-boost period. It was also tested early in the mission on 1998-01-13.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal control of spacecraft used in the pre-boost period</td>
</tr>
<tr>
<td>1</td>
<td>Contingency mode control used in the post-boost period</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

TRMMYawUpdateS (1-byte integer, array size: nscan3):

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

TRMMqac (1-byte integer, array size: nscan3):

The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

ephemerisUsed (1-byte char, array size: dim10 x nscan3):
The ephemeris source used to geolocate the swath. Special values are defined as:

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>255</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

primaryHeader (Group in S3)

version (1-byte integer, array size: nscan3):

type (1-byte integer, array size: nscan3):
secHeaderFlag (1-byte integer, array size: nscan3):

APIID (2-byte integer, array size: nscan3):

sequenceFlag (1-byte integer, array size: nscan3):

packetSequenceCount (2-byte integer, array size: nscan3):

packetLength (2-byte unsigned integer, array size: nscan3):

navigation (Group in S3)

scPos (4-byte float, array size: XYZ x nscan3):
The position vector(m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan3):
The velocity vector ($ms^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan3):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan3):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan3):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan3):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value
**scAttRollGeoc** (4-byte float, array size: nscan3):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan3):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time.
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan3):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time.
Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan3):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan3):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time.
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan3):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time.
Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value
greenHourAng (4-byte float, array size: nscan3):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan3):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan3):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

sunData (Group in S3)

solarBetaAngle (4-byte float, array size: nscan3):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan3):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan3):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan3):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan3):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow,
based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:

-9999.9 Missing value

**orbitRate** (4-byte float, array size: nscan3):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:

-9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan3):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:

-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan3):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:

-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npixelv3 x nscan3):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:

-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npixelv3 x nscan3):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: npixelv3 x nscan3):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: npixelv3 x nscan3):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npixelv3 x nscan3):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like)
5.2 1ATMI - TMI unpacked packet data

sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**moonVectorInstFrame** (4-byte float, array size: TMIxyz x nscan3):
The x, y, z components of the moon vector in the TMI instrument coordinate system.
Values are in counts. Special values are defined as:
-9999.9 Missing value

**earthView** (2-byte unsigned integer, array size: nchannel3 x npixelev3 x nscan3):

Earth view counts.

Special values are defined as:
0 Missing value

**hotLoad** (2-byte unsigned integer, array size: nchannel3 x npixelht3 x nscan3):

Hot load counts.

Special values are defined as:
0 Missing value

**coldSky** (2-byte unsigned integer, array size: nchannel3 x npixelcs3 x nscan3):

Cold sky counts.

Special values are defined as:
0 Missing value

**S4** (Swath)

**S4_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2
data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S4)
A UTC time associated with the scan.
Year (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

ephemerisUsed (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
255 Missing value

Latitude (4-byte float, array size: nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**primaryHeader** (Group in S4)

*version* (1-byte integer, array size: nscan4):

*type* (1-byte integer, array size: nscan4):

*secHeaderFlag* (1-byte integer, array size: nscan4):

*APID* (2-byte integer, array size: nscan4):

*sequenceFlag* (1-byte integer, array size: nscan4):

*packetSequenceCount* (2-byte integer, array size: nscan4):

*packetLength* (2-byte unsigned integer, array size: nscan4):

*hotLoadTemperature1* (2-byte unsigned integer, array size: nscan2):
Hot Load Thermister Count Values are in count. Special values are defined as:
- 65535 Missing value

*hotLoadTemperature2* (2-byte unsigned integer, array size: nscan2):
Hot Load Thermister Count Values are in count. Special values are defined as:
- 65535 Missing value

*hotLoadTemperature3* (2-byte unsigned integer, array size: nscan2):
Hot Load Thermister Count Values are in count. Special values are defined as:
- 65535 Missing value

*posBridgeVolt* (2-byte unsigned integer, array size: nscan2):
Positive Bridge Voltage Count. Values are in count. Special values are defined as:
- 65535 Missing value

*nearZeroVolt* (2-byte unsigned integer, array size: nscan2):
Near zero voltage of hot load bridge reference. Values are in count. Special values are defined as:
- 65535 Missing value

*gain* (1-byte char, array size: nchannelall x nscan2):
Gain for each channel. Special values are defined as:
- 255 Missing value
TMIHKPACKET (Group in S4)

baptaMotorCurrent (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

momentumUnbalance (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

spu28vSecondary (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

spu16vSecondary (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

spu6vSecondary (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

spu15vPostRegulatorOutput (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

spu5vDigPostRegulatorOutput (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

spu5vAnaPostRegulatorOutput (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

spare (2-byte unsigned integer, array size: dim6 x nscan4):
Special values are defined as:
  0  Missing value

conditioningReferenceANearFullScale (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

conditioningReferenceANearZeroScale (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

powerSupplyShelfTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value
5.2 1ATMI - TMI unpacked packet data

baptaMotorTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

baptaForwardBearingTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

aftMountingPlaceTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

spuTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

bceTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

admTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

admDeploymentStatusA (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

receiverShelfTemperature (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

droShelfTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

topRadiatorTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

admDeploymentStatusB (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value

spare2 (2-byte unsigned integer, array size: dim2 x nscan4):
Special values are defined as:
  0   Missing value

conditioningReferenceBNearFullScale (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0   Missing value
conditioningReferenceBNearZeroScale (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
  0  Missing value

spare3 (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

receiverCmdStatus (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

spinupCmdStatus (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

spareCmd1Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

spareCmd2Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

clockSelectCmdStatus (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

spareCmd3Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

spareCmd4Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

spareCmd5Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

C Structure Header file:

```c
#ifndef _TK_1ATMI_H_
#define _TK_1ATMI_H_

#ifndef _L1ATMI_S4_TMIHKPACKET_
#define _L1ATMI_S4_TMIHKPACKET_

typedef struct {
    unsigned short baptaMotorCurrent;
```
unsigned short momentumUnbalance;
unsigned short spu28vSecondary;
unsigned short spu16vSecondary;
unsigned short spu6vSecondary;
unsigned short spu15vPostRegulatorOutput;
unsigned short spu5vDigPostRegulatorOutput;
unsigned short spu5vAnaPostRegulatorOutput;
unsigned short spare[6];
unsigned short conditioningReferenceANearFullScale;
unsigned short conditioningReferenceANearZeroScale;
unsigned short powerSupplyShelfTemp;
unsigned short baptaMotorTemp;
unsigned short baptaForwardBearingTemp;
unsigned short aftMountingPlaceTemp;
unsigned short spuTemp;
unsigned short bceTemp;
unsigned short admTemp;
unsigned short admDeploymentStatusA;
unsigned short receiverShelfTemperature;
unsigned short droShelfTemp;
unsigned short topRadiatorTemp;
unsigned short admDeploymentStatusB;
unsigned short spare2[2];
unsigned short conditioningReferenceBNearFullScale;
unsigned short conditioningReferenceBNearZeroScale;
unsigned char spare3;
unsigned char receiverCmdStatus;
unsigned char spinupCmdStatus;
unsigned char spareCmd1Status;
unsigned char spareCmd2Status;
unsigned char clockSelectCmdStatus;
unsigned char spareCmd3Status;
unsigned char spareCmd4Status;
unsigned char spareCmd5Status;
} L1ATMI_S4_TMIHKPACKET;

#endif

#ifndef _L1ATMI_S4_
define _L1ATMI_S4_

typedef struct {
    SCANTIME ScanTime;

    unsigned short momentumUnbalance;
    unsigned short spu28vSecondary;
    unsigned short spu16vSecondary;
    unsigned short spu6vSecondary;
    unsigned short spu15vPostRegulatorOutput;
    unsigned short spu5vDigPostRegulatorOutput;
    unsigned short spu5vAnaPostRegulatorOutput;
    unsigned short spare[6];
    unsigned short conditioningReferenceANearFullScale;
    unsigned short conditioningReferenceANearZeroScale;
    unsigned short powerSupplyShelfTemp;
    unsigned short baptaMotorTemp;
    unsigned short baptaForwardBearingTemp;
    unsigned short aftMountingPlaceTemp;
    unsigned short spuTemp;
    unsigned short bceTemp;
    unsigned short admTemp;
    unsigned short admDeploymentStatusA;
    unsigned short receiverShelfTemperature;
    unsigned short droShelfTemp;
    unsigned short topRadiatorTemp;
    unsigned short admDeploymentStatusB;
    unsigned short spare2[2];
    unsigned short conditioningReferenceBNearFullScale;
    unsigned short conditioningReferenceBNearZeroScale;
    unsigned char spare3;
    unsigned char receiverCmdStatus;
    unsigned char spinupCmdStatus;
    unsigned char spareCmd1Status;
    unsigned char spareCmd2Status;
    unsigned char clockSelectCmdStatus;
    unsigned char spareCmd3Status;
    unsigned char spareCmd4Status;
    unsigned char spareCmd5Status;
} L1ATMI_S4_TMIHKPACKET;

#endif

#ifndef _L1ATMI_S4_
define _L1ATMI_S4_

typedef struct {
    SCANTIME ScanTime;

    unsigned short momentumUnbalance;
    unsigned short spu28vSecondary;
    unsigned short spu16vSecondary;
    unsigned short spu6vSecondary;
    unsigned short spu15vPostRegulatorOutput;
    unsigned short spu5vDigPostRegulatorOutput;
    unsigned short spu5vAnaPostRegulatorOutput;
    unsigned short spare[6];
    unsigned short conditioningReferenceANearFullScale;
    unsigned short conditioningReferenceANearZeroScale;
    unsigned short powerSupplyShelfTemp;
    unsigned short baptaMotorTemp;
    unsigned short baptaForwardBearingTemp;
    unsigned short aftMountingPlaceTemp;
    unsigned short spuTemp;
    unsigned short bceTemp;
    unsigned short admTemp;
    unsigned short admDeploymentStatusA;
    unsigned short receiverShelfTemperature;
    unsigned short droShelfTemp;
    unsigned short topRadiatorTemp;
    unsigned short admDeploymentStatusB;
    unsigned short spare2[2];
    unsigned short conditioningReferenceBNearFullScale;
    unsigned short conditioningReferenceBNearZeroScale;
    unsigned char spare3;
    unsigned char receiverCmdStatus;
    unsigned char spinupCmdStatus;
    unsigned char spareCmd1Status;
    unsigned char spareCmd2Status;
    unsigned char clockSelectCmdStatus;
    unsigned char spareCmd3Status;
    unsigned char spareCmd4Status;
    unsigned char spareCmd5Status;
} L1ATMI_S4_TMIHKPACKET;
unsigned char ephemerisUsed[10];
float Latitude;
float Longitude;
PRIMARYHEADER primaryHeader;
unsigned short hotLoadTemperature1;
unsigned short hotLoadTemperature2;
unsigned short hotLoadTemperature3;
unsigned short posBridgeVolt;
unsigned short nearZeroVolt;
unsigned char gain[9];
L1ATMI_S4_TMIHKPACKET TMIHKPACKET;
}
L1ATMI_S4;
#endif

#ifndef _L1ATMI_S3_SUNDATA_
#define _L1ATMI_S3_SUNDATA_

typedef struct {
  float solarBetaAngle;
  float phaseFromOrbitMidnight;
  float sunEarthSeparation;
  float earthAngularRadius;
  float phaseOfEclipseExit;
  float orbitRate;
  float timeSinceEclipseEntry;
  float sunVectorInBodyFrame[3];
} L1ATMI_S3_SUNDATA;
#endif

#ifndef _L1ATMI_S3_SCANSTATUS_
#define _L1ATMI_S3_SCANSTATUS_

typedef struct {
  unsigned char dataQuality;
  unsigned char missing;
  unsigned char modeStatus;
  short geoError;
  short geoWarning;
  short SCorientation;
  short pointingStatus;
  signed char acsModeMidScan;
}
signed char targetSelectionMidScan;
unsigned char tmiIsStatus;
double FractionalGranuleNumber;
short attDetermSource;
signed char TRMMcontMode;
signed char TRMMyawUpdateS;
signed char TRMMqac;
} L1ATMI_S3_SCANSTATUS;

#endif

#ifndef _L1ATMI_S3_
define _L1ATMI_S3_

typedef struct {
  SCANTIME ScanTime;
  float Latitude[208];
  float Longitude[208];
  L1ATMI_S3_SCANSTATUS scanStatus;
  unsigned char ephemerisUsed[10];
  PRIMARYHEADER primaryHeader;
  NAVIGATION navigation;
  L1ATMI_S3_SUNDATA sunData;
  float incidenceAngle[208];
  float satAzimuthAngle[208];
  float solarZenAngle[208];
  float solarAzimuthAngle[208];
  float sunGlintAngle[208];
  float moonVectorInstFrame[3];
  unsigned short earthView[208][2];
  unsigned short hotLoad[16][2];
  unsigned short coldSky[16][2];
} L1ATMI_S3;
#endif

#ifndef _L1ATMI_S2_SUNDATA_
define _L1ATMI_S2_SUNDATA_

typedef struct {
  float solarBetaAngle;
  float phaseFromOrbitMidnight;
  float sunEarthSeparation;
} L1ATMI_S2_SUNDATA;
float earthAngularRadius;
float phaseOfEclipseExit;
float orbitRate;
float timeSinceEclipseEntry;
float sunVectorInBodyFrame[3];
} L1ATMI_S2_SUNDATA;

#endif

#ifndef _L1ATMI_S2_SCANSTATUS_
#define _L1ATMI_S2_SCANSTATUS_

typedef struct {
    unsigned char dataQuality;
    unsigned char missing;
    unsigned char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    unsigned char tmiIsStatus;
    double FractionalGranuleNumber;
    short attDetermSource;
    signed char TRMMcontMode;
    signed char TRMMyawUpdateS;
    signed char TRMMqac;
} L1ATMI_S2_SCANSTATUS;

#endif

#ifndef _L1ATMI_S2_
#define _L1ATMI_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[104];
    float Longitude[104];
    L1ATMI_S2_SCANSTATUS scanStatus;
    unsigned char ephemerisUsed[10];
    PRIMARYHEADER primaryHeader;
    NAVIGATION navigation;
} L1ATMI_S2_SUNDATA;
L1ATMI_S2_SUNDATA sunData;
float incidenceAngle[104];
float satAzimuthAngle[104];
float solarZenAngle[104];
float solarAzimuthAngle[104];
float sunGlintAngle[104];
float moonVectorInstFrame[3];
unsigned short earthView[104][5];
unsigned short hotLoad[8][5];
unsigned short coldSky[8][5];
} L1ATMI_S2;
#endif

#ifndef _L1ATMI_S1_SUNDATA_
#define _L1ATMI_S1_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1ATMI_S1_SUNDATA;
#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
} L1ATMI_S1_SUNDATA;
#endif
float scAttRollGeod;
float scAttPitchGeod;
float scAttYawGeod;
float greenHourAng;
double timeMidScan;
double timeMidScanOffset;
} NAVIGATION;
#endif

#ifndef _PRIMARYHEADER_
#define _PRIMARYHEADER_

typedef struct {
  signed char version;
  signed char type;
  signed char secHeaderFlag;
  short APID;
  signed char sequenceFlag;
  short packetSequenceCount;
  unsigned short packetLength;
} PRIMARYHEADER;
#endif

#ifndef _L1ATMI_S1_SCANSTATUS_
#define _L1ATMI_S1_SCANSTATUS_

typedef struct {
  unsigned char dataQuality;
  unsigned char missing;
  unsigned char modeStatus;
  short geoError;
  short geoWarning;
  short SCorientation;
  short pointingStatus;
  signed char acsModeMidScan;
  signed char targetSelectionMidScan;
  unsigned char tmiIsStatus;
  double FractionalGranuleNumber;
  short attDetermSource;
  signed char TRMMcontMode;
  signed char TRMMyawUpdateS;
signed char TRMMqac;
} L1ATMI_S1SCANSTATUS;

#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
  short Year;
  signed char Month;
  signed char DayOfMonth;
  signed char Hour;
  signed char Minute;
  signed char Second;
  short MilliSecond;
  short DayOfYear;
  double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L1ATMI_S1_
#define _L1ATMI_S1_

typedef struct {
  SCANTIME ScanTime;
  float Latitude[104];
  float Longitude[104];
  L1ATMI_S1SCANSTATUS scanStatus;
  unsigned char ephemerisUsed[10];
  PRIMARYHEADER primaryHeader;
  NAVIGATION navigation;
  L1ATMI_S1SUNDATA sunData;
  float incidenceAngle[104][2];
  float satAzimuthAngle[104];
  float solarZenAngle[104];
  float solarAzimuthAngle[104];
  float sunGlintAngle[104];
  float moonVectorInstFrame[3];
  unsigned short earthView[104][2];
  unsigned short hotLoad[8][2];
  unsigned short coldSky[8][2];
typedef struct {
    L1ATMI_S1 S1;
    L1ATMI_S2 S2;
    L1ATMI_S3 S3;
    L1ATMI_S4 S4;
} L1ATMI_SWATHS;

typedef struct {
    float swathOffsets[3][3];
    float chnl10AngleDiff[1];
} L1ATMI_TMI1AHEADER;

Fortran Structure Header file:

STRUCTURE /L1ATMI_S4_TMIHKPACKET/
    INTEGER*2 baptaMotorCurrent
    INTEGER*2 momentumUnbalance
    INTEGER*2 spu28vSecondary
    INTEGER*2 spu16vSecondary
    INTEGER*2 spu6vSecondary
    INTEGER*2 spu15vPostRegulatorOutput
    INTEGER*2 spu5vDigPostRegulatorOutput
    INTEGER*2 spu5vAnaPostRegulatorOutput
    INTEGER*2 spare(6)
    INTEGER*2 conditioningReferenceANearFullScale
    INTEGER*2 conditioningReferenceANearZeroScale
    INTEGER*2 powerSupplyShelfTemp
INTEGER*2 baptaMotorTemp
INTEGER*2 baptaForwardBearingTemp
INTEGER*2 aftMountingPlaceTemp
INTEGER*2 spuTemp
INTEGER*2 bceTemp
INTEGER*2 admTemp
INTEGER*2 admDeploymentStatusA
INTEGER*2 receiverShelfTemperature
INTEGER*2 droShelfTemp
INTEGER*2 topRadiatorTemp
INTEGER*2 admDeploymentStatusB
INTEGER*2 spare2(2)
INTEGER*2 conditioningReferenceBNearFullScale
INTEGER*2 conditioningReferenceBNearZeroScale
CHARACTER spare3
CHARACTER receiverCmdStatus
CHARACTER spinupCmdStatus
CHARACTER spareCmd1Status
CHARACTER spareCmd2Status
CHARACTER clockSelectCmdStatus
CHARACTER spareCmd3Status
CHARACTER spareCmd4Status
CHARACTER spareCmd5Status
END STRUCTURE

STRUCTURE /L1ATMI_S4/
  RECORD /SCANTIME/ ScanTime
  CHARACTER ephemerisUsed(10)
  REAL*4 Latitude
  REAL*4 Longitude
  RECORD /PRIMARYHEADER/ primaryHeader
  INTEGER*2 hotLoadTemperature1
  INTEGER*2 hotLoadTemperature2
  INTEGER*2 hotLoadTemperature3
  INTEGER*2 posBridgeVolt
  INTEGER*2 nearZeroVolt
  CHARACTER gain(9)
  RECORD /L1ATMI_S4_TMIHKPACKET/ TMIHKPACKET
END STRUCTURE

STRUCTURE /L1ATMI_S3_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
REAL*4 sunEarthSeparation
REAL*4 earthAngularRadius
REAL*4 phaseOfEclipseExit
REAL*4 orbitRate
REAL*4 timeSinceEclipseEntry
REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1ATMI_S3_SCANSTATUS/
   CHARACTER dataQuality
   CHARACTER missing
   CHARACTER modeStatus
   INTEGER*2 geoError
   INTEGER*2 geoWarning
   INTEGER*2 SCorientation
   INTEGER*2 pointingStatus
   BYTE acsModeMidScan
   BYTE targetSelectionMidScan
   CHARACTER tmiIsStatus
   REAL*8 FractionalGranuleNumber
   INTEGER*2 attDetermSource
   BYTE TRMMcontMode
   BYTE TRMMyawUpdateS
   BYTE TRMMqac
END STRUCTURE

STRUCTURE /L1ATMI_S3/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(208)
   REAL*4 Longitude(208)
   RECORD /L1ATMI_S3_SCANSTATUS/ scanStatus
   CHARACTER ephemerisUsed(10)
   RECORD /PRIMARYHEADER/ primaryHeader
   RECORD /NAVIGATION/ navigation
   RECORD /L1ATMI_S3_SUNDATA/ sunData
   REAL*4 incidenceAngle(208)
   REAL*4 satAzimuthAngle(208)
   REAL*4 solarZenAngle(208)
   REAL*4 solarAzimuthAngle(208)
   REAL*4 sunGlintAngle(208)
   REAL*4 moonVectorInstFrame(3)
   INTEGER*2 earthView(2,208)
   INTEGER*2 hotLoad(2,16)
INTEGER*2 coldSky(2,16)
END STRUCTURE

STRUCTURE /L1ATMI_S2_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
  REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1ATMI_S2_SCANSTATUS/
  CHARACTER dataQuality
  CHARACTER missing
  CHARACTER modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  CHARACTER tmiIsStatus
  REAL*8 FractionalGranuleNumber
  INTEGER*2 attDetermSource
  BYTE TRMcontMode
  BYTE TRMMyawUpdateS
  BYTE TRMMqac
END STRUCTURE

STRUCTURE /L1ATMI_S2/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(104)
  REAL*4 Longitude(104)
  RECORD /L1ATMI_S2_SCANSTATUS/ scanStatus
  CHARACTER ephemerisUsed(10)
  RECORD /PRIMARYHEADER/ primaryHeader
  RECORD /NAVIGATION/ navigation
  RECORD /L1ATMI_S2_SUNDATA/ sunData
  REAL*4 incidenceAngle(104)
  REAL*4 satAzimuthAngle(104)
REAL*4 solarZenAngle(104)
REAL*4 solarAzimuthAngle(104)
REAL*4 sunGlintAngle(104)
REAL*4 moonVectorInstFrame(3)
INTEGER*2 earthView(5,104)
INTEGER*2 hotLoad(5,8)
INTEGER*2 coldSky(5,8)
END STRUCTURE

STRUCTURE /L1ATMI_S1_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
  REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
  REAL*4 scAlt
  REAL*4 dprAlt
  REAL*4 scAttRollGeoc
  REAL*4 scAttPitchGeoc
  REAL*4 scAttYawGeoc
  REAL*4 scAttRollGeod
  REAL*4 scAttPitchGeod
  REAL*4 scAttYawGeod
  REAL*4 greenHourAng
  REAL*8 timeMidScan
  REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /PRIMARYHEADER/
  BYTE version
  BYTE type
  BYTE secHeaderFlag
  INTEGER*2 APID
BYTE sequenceFlag
INTEGER*2 packetSequenceCount
INTEGER*2 packetLength
END STRUCTURE

STRUCTURE /L1ATMI_S1_SCANSTATUS/
  CHARACTER dataQuality
  CHARACTER missing
  CHARACTER modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  CHARACTER tmiIsStatus
  REAL*8 FractionalGranuleNumber
  INTEGER*2 attDetermSource
  BYTE TRMMcontMode
  BYTE TRMMyawUpdateS
  BYTE TRMMqac
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1ATMI_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(104)
  REAL*4 Longitude(104)
  RECORD /L1ATMI_S1_SCANSTATUS/ scanStatus
  CHARACTER ephemerisUsed(10)
  RECORD /PRIMARYHEADER/ primaryHeader
  RECORD /NAVIGATION/ navigation
RECORD /L1ATMI_S1_SUNDATA/ sunData
REAL*4 incidenceAngle(2,104)
REAL*4 satAzimuthAngle(104)
REAL*4 solarZenAngle(104)
REAL*4 solarAzimuthAngle(104)
REAL*4 sunGlintAngle(104)
REAL*4 moonVectorInstFrame(3)
INTEGER*2 earthView(2,104)
INTEGER*2 hotLoad(2,8)
INTEGER*2 coldSky(2,8)
END STRUCTURE

STRUCTURE /L1ATMI_SWATHS/
    RECORD /L1ATMI_S1/ S1;
    RECORD /L1ATMI_S2/ S2;
    RECORD /L1ATMI_S3/ S3;
    RECORD /L1ATMI_S4/ S4;
END STRUCTURE

STRUCTURE /L1ATMI_TMI1AHEADER/
    REAL*4 swathOffsets(3,3)
    REAL*4 chnl10AngleDiff(1)
END STRUCTURE

5.3 1AVIRS - VIRS unpacked packet data

1AVIRS contains unpacked packet data from VIRS science data from the VIRS instrument flown on the TRMM satellite. There are 2 swaths. Swath S1 has 5 science channels; Swath S2 has Housekeeping.

    The S1 channels are:
    0.63 micrometers
    1.6 micrometers
    3.75 micrometers
    10.8 micrometers
    12.0 micrometers

S2 has VIRS housekeeping.

We define the spacecraft axis v, used in the definition of the variable SOrientation...
RELATION BETWEEN THE SWATHS: Only Swath S1 has science data.

Dimension definitions:
	nsoparm 3 Number of swath offset parameters: cone angle, start angle, first pixel time.

np2 2 TBD.

ns swath 1 Number of swaths.

VH 2 Number of polarizations.

nscan1 var Typical number of Swath S1 scans in the granule.

nchannel1 5 Number of Swath S1 channels.

npixelev1 261 Number of earth view pixels in one scan.

npixelht1 8 Number of hot load pixels in one scan.

npixelcs1 8 Number of cold sky pixels in one scan.

nscan2 var Typical number of Swath S2 scans in the granule.

nchannelall 5 Number of all channels.

dim2 2 Number.

dim3 3 Number.

dim4 4 Number.

dim5 5 Number.

dim6 6 Number.

dim7 7 Number.

dim8 8 Number.

dim9 9 Number.

dim10 10 Number.

dim11 11 Number.

dim12 12 Number.

VIRSxyz 3 x, y, z components in VIRS instrument coordinate system.

nsample 2 Number of samples in blackbody, spaceview, solarDiffuser.

Figure 73 through Figure 89 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 73: Data Format Structure for 1AVIRS, VIRS unpacked packet data
Figure 74: Data Format Structure for 1AVIRS, S1
Figure 75: Data Format Structure for 1AVIRS, S2

Figure 76: Data Format Structure for 1AVIRS, S1, ScanTime
### 5.3 1AVIRS - VIRS unpacked packet data

**Figure 77: Data Format Structure for 1AVIRS, S1, scanStatus**

- **dataQuality**: 1 byte, Array: nscan1
- **missing**: 1 byte, Array: nscan1
- **modeStatus**: 1 byte, Array: nscan1
- **geoError**: 2 bytes, Array: nscan1
- **geoWarning**: 2 bytes, Array: nscan1
- **SCorientation**: 2 bytes, Array: nscan1
- **pointingStatus**: 2 bytes, Array: nscan1
- **acsModeMidScan**: 1 byte, Array: nscan1
- **targetSelectionMidScan**: 1 byte, Array: nscan1
- **virsInstS**: 1 byte, Array: nscan1
- **virsMode**: 1 byte, Array: nscan1
- **virsAbnCon**: 1 byte, Array: nscan1
- **FractionalGranuleNumber**: 8 bytes, Array: nscan1
- **attDetermSource**: 2 bytes, Array: nscan1
- **TRMMcontMode**: 1 byte, Array: nscan1
- **TRMMyawUpdateS**: 1 byte, Array: nscan1
- **TRMMqac**: 1 byte, Array: nscan1

**Figure 78: Data Format Structure for 1AVIRS, S1, primaryHeader**

- **version**: 1 byte, Array: nscan1
- **type**: 1 byte, Array: nscan1
- **secHeaderFlag**: 1 byte, Array: nscan1
- **APID**: 2 bytes, Array: nscan1
- **sequenceFlag**: 1 byte, Array: nscan1
- **packetSequenceCount**: 2 bytes, Array: nscan1
- **packetLength**: 2 bytes, Array: nscan1
Figure 79: Data Format Structure for 1AVIRS, S1, navigation

Figure 80: Data Format Structure for 1AVIRS, S1, solarCal

Figure 81: Data Format Structure for 1AVIRS, S1, sunData
### 5.3 1AVIRS - VIRS unpacked packet data

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOLARCALDOORDRIVERSELECT</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SCANDRIVELOCK</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SOLARCALDOORCLOSED</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SOLARCALDOOROPEN</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>RADCOOLERDOORDRIVERSELECT</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>RADCoolerDOOROUTGAS</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>RADCOOLERDOORCLOSED</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>RADCOOLERDOOROPEN</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SOLARPANELSHIELDDOORDRIVERSELECT</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SCANNMIRRORSIDE</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SOLARPANELSHIELDDOORCLOSED</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SOLARPANELSHIELDDOOROPEN</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SOLARPANELSHIELDDOORDRIVERSTATUS</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>LWIRGAIN</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SAFEHOLDSHUTTERBACKUPACTOR</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SAFEHOLDSHUTTEROPEN</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>SCANDRIVECIRCUISELECT</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>PRIMARYSCANDRIVEPOWER</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>PRIMARY10WOPERATIONALHEATER</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>PRIMARY5WOPERATIONALHEATER</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>INFORMATIONSYSTEMEVENTPRESENT</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>REDUNDANTSCANDRIVEPOWER</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>REDUNDANT10WOPERATIONALHEATER</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
<tr>
<td>REDUNDANT5WOPERATIONALHEATER</td>
<td>1 byte</td>
<td>nscan1</td>
</tr>
</tbody>
</table>

continued on next figure
Figure 83: Data Format Structure for 1AVIRS, S1, virsPacketHeader
Figure 84: Data Format Structure for 1AVIRS, S2, ScanTime

Figure 85: Data Format Structure for 1AVIRS, S2, primaryHeader
FileHeader (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

AlgorithmRuntimeInfo (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

S1 (Swath)

S1_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S1)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value
5.3 1AVIRS - VIRS unpacked packet data

virsPacketHeader

- SOLARCALDOORDRIVERSELECT 1 byte
  Array: nscan2
- SCANDRIVELOCK 1 byte
  Array: nscan2
- SOLARCALDOORCLOSED 1 byte
  Array: nscan2
- SOLARCALDOOROPEN 1 byte
  Array: nscan2
- RADCOOLERDOORDRIVERSELECT 1 byte
  Array: nscan2
- RADCOOLERDOOROUTGAS 1 byte
  Array: nscan2
- RADCOOLERDOORCLOSED 1 byte
  Array: nscan2
- RADCOOLERDOOROPEN 1 byte
  Array: nscan2
- SOLARPANELSHIELDDRIVERSELECT 1 byte
  Array: nscan2
- SCANMIRRORSIDE 1 byte
  Array: nscan2
- SOLARPANELSHIELDDOORCLOSED 1 byte
  Array: nscan2
- SOLARPANELSHIELDDOOROPEN 1 byte
  Array: nscan2
- SOLARPANELSHIELDDRIVERSTATUS 1 byte
  Array: nscan2
- LWIRGAIN 1 byte
  Array: nscan2
- SAFEHOLDSHUTTERBACKUPACTOR 1 byte
  Array: nscan2
- SAFEHOLDSHUTTEROPEN 1 byte
  Array: nscan2
- SCANDRIVECIRCUISELECT 1 byte
  Array: nscan2
- PRIMARYSCANDRIVEPOWER 1 byte
  Array: nscan2
- PRIMARY10WOPERATIONALHEATER 1 byte
  Array: nscan2
- PRIMARY5WOPERATIONALHEATER 1 byte
  Array: nscan2
- INFORMATIONEVENTPRESENT 1 byte
  Array: nscan2
- REDUNDANTSCANDRIVEPOWER 1 byte
  Array: nscan2
- REDUNDANT10WOPERATIONALHEATER 1 byte
  Array: nscan2
- REDUNDANT5WOPERATIONALHEATER 1 byte
  Array: nscan2

continued on next figure

•

•

•
continued from last figure

Figure 87: Data Format Structure for 1AVIRS, S2, virsPacketHeader
5.3 1AVIRS - VIRS unpacked packet data

VIRSKHFPACKET

- PRIMARYSCANDRIVESERVOPHASEERROR 4 bytes Array: nscan2
- TELEMETRYBOARDTEMP 4 bytes Array: nscan2
- PRIMARYSCANDRIVEMOTORCURRENT 4 bytes Array: nscan2
- REDUNDANTSCANDRIVEMOTORCURRENT 4 bytes Array: nscan2
- COOLERINTERMEDIATESTATEGETEMP 4 bytes Array: nscan2
- COOLERCOLDESTATEGETEMP 4 bytes Array: nscan2
- FPATEMPCONTROLPOWER 4 bytes Array: nscan2
- PRIMARYAPOWERSUPPYTEMP 4 bytes Array: nscan2
- REDUNDANTBPOWERSUPPYTEMP 4 bytes Array: nscan2
- MOTORENCODERTEMP 4 bytes Array: nscan2
- COOLER MountRINGTEMP 4 bytes Array: nscan2
- PRIMARYBLACKBODYTEMP 4 bytes Array: nscan2
- REDUNDANTBLACKBODYTEMP 4 bytes Array: nscan2
- PRIMARYPSSECONDARYVOLTAGEPOS28 4 bytes Array: nscan2
- PRIMARYPSVOLTAGEPOS16 4 bytes Array: nscan2
- PRIMARYPSVOLTAGENEG16 4 bytes Array: nscan2
- REDUNDANTSCANDRIVESERVOPHASEERROR 4 bytes Array: nscan2
- PRIMARYPSVOLTAGEPOS12 4 bytes Array: nscan2
- PRIMARYPSVOLTAGENEG12 4 bytes Array: nscan2
- PRIMARYPSVOLTAGEPOS8 4 bytes Array: nscan2
- PRIMARYPSVOLTAGENEG8 4 bytes Array: nscan2
- REDUNDANTPSSECONDARYVOLTAGEPOS28 4 bytes Array: nscan2
- REDUNDANTPSVOLTAGEPOS15 4 bytes Array: nscan2
continued from last figure

![Diagram](image)

Figure 89: Data Format Structure for 1AVIRS, S2, VIRSHKPACKET
Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixele1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixele1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S1)

dataQuality (1-byte char, array size: nscan1):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
</tbody>
</table>
6  modeStatus is not normal
7  QAC errors associated with this scan

**missing** (1-byte char, array size: nscan1):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte char, array size: nscan1):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation is not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine instrument status</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan1):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit $7 = 0$ and all of these flags are also 0.

Bit Meaning if bit = 1
0 Latitude limit exceeded for viewed pixel locations
1 Negative scan time, invalid input
2 Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan1):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits = 0 the unsigned integer value is $2^{**i}$):

Bit Meaning if bit = 1
0 Ephemeral Gap Interpolated
1 Attitude Gap Interpolated
2 Attitude jump/discontinuity
3 Attitude out of range
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

SCorientation (2-byte integer, array size: nscan1):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8002</td>
<td>Yaw turn in progress</td>
</tr>
<tr>
<td>-8003</td>
<td>Deep Space Calibration in progress</td>
</tr>
<tr>
<td>-8004</td>
<td>Non-nominal pointing other than above</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan1):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal ACS mode (4) for mission science</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal ACS mode</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan1):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>
**targetSelectionMidScan** (1-byte integer, array size: nscan1):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**virsInstS** (1-byte integer, array size: nscan1):

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Day (no calibration occurring)</td>
</tr>
<tr>
<td>1</td>
<td>Night</td>
</tr>
<tr>
<td>2</td>
<td>Monitor Scan Stability</td>
</tr>
<tr>
<td>3</td>
<td>Day with Calibration</td>
</tr>
</tbody>
</table>

**virsMode** (1-byte integer, array size: nscan1):

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>mission mode</td>
</tr>
<tr>
<td>1</td>
<td>safehold mode</td>
</tr>
<tr>
<td>2</td>
<td>outgas mode</td>
</tr>
<tr>
<td>3</td>
<td>activation mode</td>
</tr>
</tbody>
</table>

**virsAbnCon** (1-byte char, array size: nscan1):
Bit 0 is the most significant bit (i.e., if bit \(i = 1\) and other bits = 0, the unsigned integer value is \(2^{8-i} - 1\)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>normal</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>scan phase error</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>normal</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>selftest error</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>normal</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>thermal data missing</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>normal</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>moon in space view</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>normal</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>H/K data drop-out suspected</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>normal</td>
</tr>
</tbody>
</table>
1  SV counts for channel 4 or 5 greater than L1B01\_MIN\_DNSV
6  0  not used
7  0  not used

**FractionalGranuleNumber** (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
- -9999.9  Missing value

**attDetermSource** (2-byte integer, array size: nscan1):

Attitude determination source.
A flag explaining how the attitude value was calculated.
Improved estimates make use of ground processing of PR science-instrument-measured roll values, Gyroscope data, and Sun Sensor 1 data. Earlier products (TRMM V7 and before) used the onboard attitudes with various corrections. Values were determined for each granule based on the data available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430 and higher</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
<tr>
<td>421</td>
<td>Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)</td>
</tr>
<tr>
<td>413</td>
<td>Reduced accuracy, sun data not available (affecting pitch)</td>
</tr>
<tr>
<td>411</td>
<td>Reduced accuracy, PR roll and sun sensor not available</td>
</tr>
<tr>
<td>300-399</td>
<td>Reduced accuracy due to various special conditions</td>
</tr>
<tr>
<td>200-299</td>
<td>Fallback to using the onboard attitude estimates with TRMM V7 corrections</td>
</tr>
<tr>
<td>-91</td>
<td>Spacecraft in safehold mode, no science data</td>
</tr>
<tr>
<td>-99</td>
<td>No data due to telemetry data gap</td>
</tr>
</tbody>
</table>

**TRMMcontMode** (1-byte integer, array size: nscan1):
The Contingency Mode Flag from telemetry indicates alternate attitude control of the spacecraft.
The nominal at-launch Attitude Control System (ACS) for TRMM used Earth horizon sensors for pitch and roll control, and the yaw was updated twice each orbit using the Sun Sensors and propagated using gyro data.
However, due to possible problems identified with the Earth Sensor Assembly (ESA) lifetime on-orbit, a contingency ACS mode was developed late in the development cycle. This mode used the Sun Sensors, magnetometers, and gyroscope data. It proved very valuable when the horizon sensors had problems with TRMM moving to the higher operating altitude (from 350 to 402.5 km) to extend the mission lifetime. Thus the contingency mode was used throughout the post-boost period. It was also tested early in the mission on 1998-01-13.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal control of spacecraft used in the pre-boost period</td>
</tr>
<tr>
<td>1</td>
<td>Contingency mode control used in the post-boost period</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMyawUpdateS (1-byte integer, array size: nscan1):**

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMqac (1-byte integer, array size: nscan1):**
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

**ephemerisUsed (1-byte char, array size: dim10 x nscan1):**
The ephemeris source used to geolocate the swath. Special values are defined as:

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>255</td>
<td>Missing value</td>
</tr>
</tbody>
</table>
primaryHeader (Group in S1)

version (1-byte integer, array size: nscan1):

type (1-byte integer, array size: nscan1):

secHeaderFlag (1-byte integer, array size: nscan1):

APID (2-byte integer, array size: nscan1):

sequenceFlag (1-byte integer, array size: nscan1):

packetSequenceCount (2-byte integer, array size: nscan1):

packetLength (2-byte unsigned integer, array size: nscan1):

navigation (Group in S1)

scPos (4-byte float, array size: XYZ x nscan1):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan1):
The velocity vector ($ms^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan1):
The geodetic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan1):
The geodetic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
scAlt (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

dprAlt (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

scAttRollGeoc (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scAttPitchGeoc (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scAttYawGeoc (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9  Missing value

scAttRollGeod (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan1):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan1):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan1):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan1):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan1):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**solarCal** (Group in S1)

**sunVecX** (8-byte float, array size: nscan1):
Solar Position (X-component) (Geocentric Inertial Coord).

**sunVecY** (8-byte float, array size: nscan1):
Solar Position (Y-component) (Geocentric Inertial Coord).

**sunVecZ** (8-byte float, array size: nscan1):
Solar Position (Z-component) (Geocentric Inertial Coord).

**sunMag** (8-byte float, array size: nscan1):
Sun-Earth Distance (m).

**sunData** (Group in S1)
5.3 1AVIRS - VIRS unpacked packet data

solarBetaAngle (4-byte float, array size: nscan1):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan1):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan1):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan1):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan1):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nscan1):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan1):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan1):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value
incidenceAngle (4-byte float, array size: npixelev1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
   -9999.9  Missing value

satAzimuthAngle (4-byte float, array size: npixelev1 x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9  Missing value

solarZenAngle (4-byte float, array size: npixelev1 x nscan1):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
   -9999.9  Missing value

solarAzimuthAngle (4-byte float, array size: npixelev1 x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9  Missing value

sunGlintAngle (4-byte float, array size: npixelev1 x nscan1):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
   -9999.9  Missing value

moonVectorInstFrame (4-byte float, array size: VIRSxyz x nscan1):
The x, y, z components of the moon vector in the VIRS instrument coordinate system. Values are in counts. Special values are defined as:
   -9999.9  Missing value

blackbody (2-byte unsigned integer, array size: nchannel1 x nsample x nscan1):
Radiance measured from VIRS onboard blackbody calibration target.

spaceview (2-byte unsigned integer, array size: nchannel1 x nsample x nscan1):
radiance measured from deep space.
solarDiffuser (2-byte unsigned integer, array size: nchannel1 x nsample x nscan1):

Reflectance measured
from VIRS onboard solar diffuser calibration target.

nightFill (1-byte char, array size: nscan1):
TBD.

earthView (2-byte unsigned integer, array size: nchannel1 x npixelev1 x nscan1):
Radiance measured from earth view.

virsPacketHeader (Group in S1)

SOLARCALDOORDRIVERSELECT (1-byte char, array size: nscan1):

SCANDRIVELOCK (1-byte char, array size: nscan1):

SOLARCALDOORCLOSED (1-byte char, array size: nscan1):

SOLARCALDOOROPEN (1-byte char, array size: nscan1):

RADCOOLERDOORDRIVERSELECT (1-byte char, array size: nscan1):
RADCOOLERDOOROUTGAS (1-byte char, array size: nscan1):

RADCOOLERDOORCLOSED (1-byte char, array size: nscan1):

RADCOOLERDOOROPEN (1-byte char, array size: nscan1):

SOLARPANELSHIELDDOORDRIVERSELECT (1-byte char, array size: nscan1):

SCANMIRRORSIDE (1-byte char, array size: nscan1):

SOLARPANELSHIELDDOORCLOSED (1-byte char, array size: nscan1):

SOLARPANELSHIELDDOOROPEN (1-byte char, array size: nscan1):

SOLARPANELSHIELDDOORDRIVERSTATUS (1-byte char, array size: nscan1):

LWIRGAIN (1-byte char, array size: nscan1):
5.3 1AVIRS - VIRS unpacked packet data

SAFEHOLDSHUTTERBACKUPACTUATOR (1-byte char, array size: nscan1):

SAFEHOLDSHUTTEROPEN (1-byte char, array size: nscan1):

SCANDRIVECIRCUISELECT (1-byte char, array size: nscan1):

PRIMARYSCANDRIVEPOWER (1-byte char, array size: nscan1):

PRIMARY10WOPERATIONALHEATER (1-byte char, array size: nscan1):

PRIMARY5WOPERATIONALHEATER (1-byte char, array size: nscan1):

INFORMATIONEVENTPRESENT (1-byte char, array size: nscan1):

REDUNDANTSCANDRIVEPOWER (1-byte char, array size: nscan1):

REDUNDANT10WOPERATIONALHEATER (1-byte char, array size: nscan1):

REDUNDANT5WOPERATIONALHEADER (1-byte char, array size: nscan1):
WARNINGEVENTPRESENT (1-byte char, array size: nscan1):

ACTIVEPROCESSOR (1-byte char, array size: nscan1):

SECONDARYPOWERSUPPLY (1-byte char, array size: nscan1):

PRIMARYPOWERSUPPLY (1-byte char, array size: nscan1):

ERROREVENTPRESENT (1-byte char, array size: nscan1):

INNERSTAGEHEATER (1-byte char, array size: nscan1):

OUTERSTAGEHEATER (1-byte char, array size: nscan1):

OUTGASENABLE (1-byte char, array size: nscan1):

SERVOLOW (1-byte char, array size: nscan1):
SERVOHIGH (1-byte char, array size: nscan1):

PRIMOUNTRIGHEATER (1-byte char, array size: nscan1):

RDTMOUNTRINGHEATER (1-byte char, array size: nscan1):

PRIMARYBLACKBODYTEMP (2-byte unsigned integer, array size: nscan1):

RDTMOUNTRINGHEATERENABLED (1-byte char, array size: nscan1):

PADDING (1-byte char, array size: nscan1):

SECONDARYBACKBODYTEMP (2-byte unsigned integer, array size: nscan1):

DATAMODE (1-byte char, array size: nscan1):

PRISOLARSHIELDACTIVATE (1-byte char, array size: nscan1):

RDTSOLARSHIELDACTIVATE (1-byte char, array size: nscan1):
PROCESSORSELFTESTSTATUS (1-byte char, array size: nscan1):

RESERVED (1-byte char, array size: nscan1):

S2 (Swath)

S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value
MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

ephemerisUsed (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
255 Missing value

Latitude (4-byte float, array size: nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

primaryHeader (Group in S2)

version (1-byte integer, array size: nscan2):

type (1-byte integer, array size: nscan2):

secHeaderFlag (1-byte integer, array size: nscan2):

APID (2-byte integer, array size: nscan2):

sequenceFlag (1-byte integer, array size: nscan2):

packetSequenceCount (2-byte integer, array size: nscan2):
packetLength (2-byte unsigned integer, array size: nscan2):

virsPacketHeader (Group in S2)

SOLARCALDOORDRIVERSELECT (1-byte char, array size: nscan2):

SCANDRIVELOCK (1-byte char, array size: nscan2):

SOLARCALDOORCLOSED (1-byte char, array size: nscan2):

SOLARCALDOOROPEN (1-byte char, array size: nscan2):

RADCOOLERDOORDRIVERSELECT (1-byte char, array size: nscan2):

RADCOOLERDOOROUTGAS (1-byte char, array size: nscan2):

RADCOOLERDOORCLOSED (1-byte char, array size: nscan2):

RADCOOLERDOOROPEN (1-byte char, array size: nscan2):
5.3 1AVIRS - VIRS unpacked packet data

SOLARPANELSHIELDDOORDRIVERSELECT (1-byte char, array size: nscan2):

SCANMIRRORSIDE (1-byte char, array size: nscan2):

SOLARPANELSHIELDDOORCLOSED (1-byte char, array size: nscan2):

SOLARPANELSHIELDDOOROPEN (1-byte char, array size: nscan2):

SOLARPANELSHIELDDOORDRIVERSTATUS (1-byte char, array size: nscan2):

LWIRGAIN (1-byte char, array size: nscan2):

SAFEHOLDSHUTTERBACKUPACTUATOR (1-byte char, array size: nscan2):

SAFEHOLDSHUTTEROPEN (1-byte char, array size: nscan2):

SCANDRIVECIRCUISELECT (1-byte char, array size: nscan2):

PRIMARYSCANDRIVEPOWER (1-byte char, array size: nscan2):
PRIMARY10WOPERATIONALHEATER (1-byte char, array size: nscan2):

PRIMARY5WOPERATIONALHEATER (1-byte char, array size: nscan2):

INFORMATIONEVENTPRESENT (1-byte char, array size: nscan2):

REDUNDANTSCANDRIVEPOWER (1-byte char, array size: nscan2):

REDUNDANT10WOPERATIONALHEATER (1-byte char, array size: nscan2):

REDUNDANT5WOPERATIONALHEADER (1-byte char, array size: nscan2):

WARNINGEVENTPRESENT (1-byte char, array size: nscan2):

ACTIVEPROCESSOR (1-byte char, array size: nscan2):

SECONDARYPOWERSUPPLY (1-byte char, array size: nscan2):
5.3 1AVIRS - VIRS unpacked packet data

PRIMARYPOWERSUPPLY (1-byte char, array size: nscan2):

ERROREVENTPRESENT (1-byte char, array size: nscan2):

INNERSTAGEHEATER (1-byte char, array size: nscan2):

OUTERSTAGEHEATER (1-byte char, array size: nscan2):

OUTGASENABLE (1-byte char, array size: nscan2):

SERVOLOW (1-byte char, array size: nscan2):

SERVOHIGH (1-byte char, array size: nscan2):

PRIMOUNTRIGHEATER (1-byte char, array size: nscan2):

RDTMOUNTRINGHEATER (1-byte char, array size: nscan2):

PRIMARYBLACKBODYTEMP (2-byte unsigned integer, array size: nscan2):
RDTMOUNTRINGHEATERENABLED (1-byte char, array size: nscan2):

PADDING (1-byte char, array size: nscan2):

SECONDARYBACKBODYTEMP (2-byte unsigned integer, array size: nscan2):

DATAMODE (1-byte char, array size: nscan2):

PRISOLARSHIELDACTIVATE (1-byte char, array size: nscan2):

RDTSOLARSHIELDACTIVATE (1-byte char, array size: nscan2):

PROCESSORSELFTESTSTATUS (1-byte char, array size: nscan2):

RESERVED (1-byte char, array size: nscan2):
VIRSHKPACKET (Group in S2)

PRIMARYSCANDRIVESERVOPHASEERROR (4-byte unsigned integer, array size: nscan2):

TELEMETRYBOARDTEMP (4-byte unsigned integer, array size: nscan2):

PRIMARYSCANDRIVEMOTORCURRENT (4-byte unsigned integer, array size: nscan2):

REDUNDANTSCANDRIVEMOTORCURRENT (4-byte unsigned integer, array size: nscan2):

COOLERINTERMEDIATESTAGETEMP (4-byte unsigned integer, array size: nscan2):

COOLERCOLDSTAGETEMP (4-byte unsigned integer, array size: nscan2):

FPATEMPCONTROLPOWER (4-byte unsigned integer, array size: nscan2):

PRIMARYAPowersUPPYTEMP (4-byte unsigned integer, array size: nscan2):
REDUNDANTBPOWERSUPPYTEMP (4-byte unsigned integer, array size: nscan2):

MOTORENCODERTEMP (4-byte unsigned integer, array size: nscan2):

COOLER_MOUNTINGRINGTEMP (4-byte unsigned integer, array size: nscan2):

PRIMARYBLACKBODYTEMP (4-byte unsigned integer, array size: nscan2):

REDUNDANTBLACKBODYTEMP (4-byte unsigned integer, array size: nscan2):

PRIMARYPSSECONDARYVOLTAGEPOS28 (4-byte unsigned integer, array size: nscan2):

PRIMARYPSVOLTAGEPOS16.5 (4-byte unsigned integer, array size: nscan2):

PRIMARYPSVOLTAGENEG16.5 (4-byte unsigned integer, array size: nscan2):

REDUNDANTSCANDRIVESERVOPHASEERROR (4-byte unsigned integer, array size: nscan2):
5.3 1AVIRS - VIRS unpacked packet data

PRIMAR YPSVOL T AGENEG\textsubscript{12} (4-byte unsigned integer, array size: nscan2):

PRIMAR YPSVOL T AGEPOS\textsubscript{8.5} (4-byte unsigned integer, array size: nscan2):

PRIMAR YPSVOL T AGEPOS\textsubscript{8.0} (4-byte unsigned integer, array size: nscan2):

PRIMAR YPSVOL T AGENEG\textsubscript{8.0} (4-byte unsigned integer, array size: nscan2):

REDUNDANTPSSECONDARYVOL T AGEPOS\textsubscript{28} (4-byte unsigned integer, array size: nscan2):

REDUNDANTPSVOL T AGEPOS\textsubscript{16.5} (4-byte unsigned integer, array size: nscan2):

REDUNDANTPSVOL T AGENEG\textsubscript{16.5} (4-byte unsigned integer, array size: nscan2):

REDUNDANTPSVOL T AGEPOS\textsubscript{12} (4-byte unsigned integer, array size: nscan2):

REDUNDANTPSVOL T AGENEG\textsubscript{12} (4-byte unsigned integer, array size: nscan2):
REDUNDANTPSVOLTAGEPOS8_5 (4-byte unsigned integer, array size: nscan2):

REDUNDANTPSSECONDARYVOLTAGEPOS8_0 (4-byte unsigned integer, array size: nscan2):

REDUNDANTPSSECONDARYVOLTAGENEG8_0 (4-byte unsigned integer, array size: nscan2):

PRIMARYPSVOLTAGEPOS12 (4-byte unsigned integer, array size: nscan2):

COMMANDSACCEPTEDCOUNTER (4-byte unsigned integer, array size: nscan2):

COMMANDSREJECTEDCOUNTER (4-byte unsigned integer, array size: nscan2):

PROCESSORBIT (4-byte unsigned integer, array size: nscan2):

BCRTBIT (4-byte unsigned integer, array size: nscan2):

NUMBEROFBADMEMORYBLOCKS (4-byte unsigned integer, array size: nscan2):
5.3 1AVIRS - VIRS unpacked packet data

**SPAREWORD** (4-byte unsigned integer, array size: nscan2):

**MOSTRECENTEVENTSEQUENCENUMBER** (4-byte unsigned integer, array size: nscan2):

**MOSTRECENTEVENTMESSAGENUMBER** (4-byte unsigned integer, array size: nscan2):

**MOSTRECENTEVENTSEQUENCENUMBER2** (4-byte unsigned integer, array size: nscan2):

**MOSTRECENTEVENTMESSAGENUMBER2** (4-byte unsigned integer, array size: nscan2):

**MOSTRECENTEVENTSEQUENCENUMBER3** (4-byte unsigned integer, array size: nscan2):

**MOSTRECENTEVENTMESSAGENUMBER3** (4-byte unsigned integer, array size: nscan2):

**SAFEHOLDSTATUS** (4-byte unsigned integer, array size: nscan2):
C Structure Header file:

#ifndef _TK_1AVIRS_H_
define _TK_1AVIRS_H_
#ifndef _L1AVIRS_S2_VIRSHKPACKET_
define _L1AVIRS_S2_VIRSHKPACKET_

typedef struct {
    unsigned int PRIMARYSCANDRIVESERVOPHASEERROR;
    unsigned int TELEMETRYBOARDTEMP;
    unsigned int PRIMARYSCANDRIVEMOTORCURRENT;
    unsigned int REDUNDANTSCANDRIVEMOTORCURRENT;
    unsigned int COOLERINTERMEDIATESTAGETEMP;
    unsigned int COOLERCOOLDSTAGETEMP;
    unsigned int FPATEMPCONTROLPOWER;
    unsigned int PRIMARYAPowersUPPYTEMP;
    unsigned int REDUNDANTBPOWERSUPPYTEMP;
    unsigned int MOTORENCODERTEMP;
    unsigned int COOLER MountINGRINGTEMP;
    unsigned int PRIMARYBLACKBODYTEMP;
    unsigned int REDUNDANTBLACKBODYTEMP;
    unsigned int PRIMARYPSSECONDARYVOLTAGEPOS28;
    unsigned int PRIMARYPSVOLTAGEPOS16_5;
    unsigned int PRIMARYPSVOLTAGENEG16_5;
    unsigned int REDUNDANTSCANDRIVESERVOPHASEERROR;
    unsigned int PRIMARYPSVOLTAGENEG12;
    unsigned int PRIMARYPSVOLTAGEPOS12;
    unsigned int PRIMARYPSVOLTAGEPOS8_5;
    unsigned int PRIMARYPSVOLTAGENEG8_0;
    unsigned int REDUNDANTPSSECONDARYVOLTAGEPOS28;
    unsigned int PRIMARYPSVOLTAGEPOS16_5;
    unsigned int PRIMARYPSVOLTAGENEG16_5;
    unsigned int REDUNDANTPSVOLTAGEPOS12;
    unsigned int PRIMARYPSVOLTAGENEG12;
    unsigned int REDUNDANTPSVOLTAGEPOS8_5;
    unsigned int REDUNDANTPSSECONDARYVOLTAGEPOS8_0;
    unsigned int PRIMARYPSVOLTAGEPOS12;
    unsigned int COMMANDSACCEPTEDCOUNTER;
    unsigned int COMMANDSREJECTEDCOUNTER;
    unsigned int PROCESSORBIT;
    unsigned int BCRTBIT;
}
unsigned int NUMBEROFBADMEMORYBLOCKS;
unsigned int SPAREWORD;
unsigned int MOSTRECENTEVENTSEQUENCENUMBER;
unsigned int MOSTRECENTEVENTMESSAGENUMBER;
unsigned int MOSTRECENTEVENTSEQUENCENUMBER2;
unsigned int MOSTRECENTEVENTMESSAGENUMBER2;
unsigned int MOSTRECENTEVENTSEQUENCENUMBER3;
unsigned int MOSTRECENTEVENTMESSAGENUMBER3;
unsigned int SAFEHOLDSTATUS;
} L1AVIRS_S2_VIRSHKPACKET;

#endif

#ifndef _L1AVIRS_S2_
#define _L1AVIRS_S2_

typedef struct {
    SCANTIME ScanTime;
    unsigned char ephemerisUsed[10];
    float Latitude;
    float Longitude;
    PRIMARYHEADER primaryHeader;
    VIRSPACKETHEADER virsPacketHeader;
    L1AVIRS_S2_VIRSHKPACKET VIRSHKPACKET;
} L1AVIRS_S2;
#endif

#ifndef _VIRSPACKETHEADER_
#define _VIRSPACKETHEADER_

typedef struct {
    unsigned char SOLARCALDOORDRIVERSELECT;
    unsigned char SCANDRIVELOCK;
    unsigned char SOLARCALDOORCLOSED;
    unsigned char SOLARCALDOOROPEN;
    unsigned char RADCOOLERDOORDRIVERSELECT;
    unsigned char RADCOOLERDOOROUTGAS;
    unsigned char RADCOOLERDOORCLOSED;
    unsigned char RADCOOLERDOOROPEN;
    unsigned char SOLARPANELSHIELDDOORDRIVERSELECT;
    unsigned char SCANMIRRORSIDE;
    unsigned char SOLARPANELSHIELDDOORCLOSED;
} VIRSPACKETHEADER;
#endif
unsigned char SOLAR_PANEL_SHIELD_DOOR_OPEN;
unsigned char SOLAR_PANEL_SHIELD_DOOR_DRIVER_STATUS;
unsigned char LWR_GAIN;
unsigned char SAFE_HOLD_SHUTTER_BACKUP_ACTUATOR;
unsigned char SAFE_HOLD_SHUTTER_OPEN;
unsigned char SCAN_DRIVE_CIRCUIT_SELECT;
unsigned char PRIMARY_SCAN_DRIVE_POWER;
unsigned char PRIMARY_10_W_OPERATIONAL_HEATER;
unsigned char PRIMARY_5_W_OPERATIONAL_HEATER;
unsigned char INFORMATION_EVENT_PRESENT;
unsigned char REDUNDANT_SCAN_DRIVE_POWER;
unsigned char REDUNDANT_10_W_OPERATIONAL_HEATER;
unsigned char REDUNDANT_5_W_OPERATIONAL_HEADER;
unsigned char WARNING_EVENT_PRESENT;
unsigned char ACTIVE_PROCESSOR;
unsigned char SECONDARY_POWER_SUPPLY;
unsigned char PRIMARY_POWER_SUPPLY;
unsigned char ERROR_EVENT_PRESENT;
unsigned char INNER_STAGE_HEATER;
unsigned char OUTER_STAGE_HEATER;
unsigned char OUTGASE_ENABLE;
unsigned char SERVO_LOW;
unsigned char SERVO_HIGH;
unsigned char PRIME_MOUNT_TRIG_HEATER;
unsigned char RDT_MOUNT_RING_HEATER;
unsigned short PRIMARY_BLACK_BODY_TEMP;
unsigned char RDT_MOUNT_RING_HEATER_ENABLE;
unsigned char Padding;
unsigned short SECONDARY_BACK_BODY_TEMP;
unsigned char DATA_MODE;
unsigned char PRISM_SOLAR_SHIELD_ACTIVATE;
unsigned char RDT_SOLAR_SHIELD_ACTIVATE;
unsigned char PROCESSOR_SELF_TEST_STATUS;
unsigned char RESERVED;
} VIR_PACKET_HEADER;

#endif

#ifdef _L1AVIRS_S1_SUN_DATA_
define _L1AVIRS_S1_SUN_DATA_

typedef struct {
    float solarBetaAngle;
}
float phaseFromOrbitMidnight;
float sunEarthSeparation;
float earthAngularRadius;
float phaseOfEclipseExit;
float orbitRate;
float timeSinceEclipseEntry;
float sunVectorInBodyFrame[3];
}
L1AVIRS_S1_SUNDATA;

#endif

#ifndef _L1AVIRS_S1_SOLARCAL_
#define _L1AVIRS_S1_SOLARCAL_

typedef struct {
    double sunVecX;
    double sunVecY;
    double sunVecZ;
    double sunMag;
} L1AVIRS_S1_SOLARCAL;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;

#endif
#ifndef _PRIMARYHEADER_
define _PRIMARYHEADER_

typedef struct {
    signed char version;
    signed char type;
    signed char secHeaderFlag;
    short APID;
    signed char sequenceFlag;
    short packetSequenceCount;
    unsigned short packetLength;
} PRIMARYHEADER;

#endif

#ifndef _L1AVIRS_S1_SCANSTATUS_
define _L1AVIRS_S1_SCANSTATUS_

typedef struct {
    unsigned char dataQuality;
    unsigned char missing;
    unsigned char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char virsInstS;
    signed char virsMode;
    unsigned char virsAbnCon;
    double FractionalGranuleNumber;
    short attDetermSource;
    signed char TRMMcontMode;
    signed char TRMYawUpdateS;
    signed char TRMqac;
} L1AVIRS_S1_SCANSTATUS;

#endif
#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L1AVIRS_S1_
define _L1AVIRS_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[261];
    float Longitude[261];
    L1AVIRS_S1_SCANSTATUS scanStatus;
    unsigned char ephemerisUsed[10];
    PRIMARYHEADER primaryHeader;
    NAVIGATION navigation;
    L1AVIRS_S1_SOLARCAL solarCal;
    L1AVIRS_S1_SUNDATA sunData;
    float incidenceAngle[261];
    float satAzimuthAngle[261];
    float solarZenAngle[261];
    float solarAzimuthAngle[261];
    float sunGlintAngle[261];
    float moonVectorInstFrame[3];
    unsigned short blackbody[2][5];
    unsigned short spaceview[2][5];
    unsigned short solarDiffuser[2][5];
    unsigned char nightFill;
    unsigned short earthView[261][5];
    VIRSPACKETHEADER virsPacketHeader;
} L1AVIRS_S1;
#endif
typedef struct {
    L1AVIRS_S1 S1;
    L1AVIRS_S2 S2;
} L1AVIRS_SWATHS;

Fortran Structure Header file:

STRUCTURE /L1AVIRS_S2_VIRSHKPACKET/
    INTEGER*4 PRIMARYSCANDRIVESERVOPHASEERROR
    INTEGER*4 TELEMETRYBOARDTEMP
    INTEGER*4 PRIMARYSCANDRIVEMOTORCURRENT
    INTEGER*4 REDUNDANTSCANDRIVEMOTORCURRENT
    INTEGER*4 COOLERINTERMEDIATESTAGETEMP
    INTEGER*4 COOLERCOLDESTAGETEMP
    INTEGER*4 FPATEMPCONTROLPOWER
    INTEGER*4 PRIMARYAPOWERSUPPYTEMP
    INTEGER*4 REDUNDANTBPOWERSUPPYTEMP
    INTEGER*4 MOTORENCODERTEMP
    INTEGER*4 COOLERMOUNTINGRINGTEMP
    INTEGER*4 PRIMARYBLACKBODYTEMP
    INTEGER*4 REDUNDANTBLACKBODYTEMP
    INTEGER*4 PRIMARYPSSECONDARYVOLTAGEPOS28
    INTEGER*4 PRIMARYPSVOLTAGEPOS16_5
    INTEGER*4 PRIMARYPSVOLTAGENEG16_5
    INTEGER*4 REDUNDANTSCANDRIVESERVOPHASEERROR
    INTEGER*4 PRIMARYPSVOLTAGENEG12
    INTEGER*4 PRIMARYPSVOLTAGEPOS8_5
    INTEGER*4 PRIMARYPSVOLTAGEPOS8_0
    INTEGER*4 PRIMARYPSVOLTAGENEG8_0
    INTEGER*4 REDUNDANTPSSECONDARYVOLTAGEPOS28
    INTEGER*4 REDUNDANTPSVOLTAGEPOS16_5
    INTEGER*4 REDUNDANTPSVOLTAGENEG16_5
    INTEGER*4 REDUNDANTPSVOLTAGEPOS12
5.3 1AVIRS - VIRS unpacked packet data

INTEGER*4 REDUNDANTPSVOLTAGENEG12
INTEGER*4 REDUNDANTPSVOLTAGEPOS8_5
INTEGER*4 REDUNDANTPSSECONDARYVOLTAGEPOS8_0
INTEGER*4 REDUNDANTPSSECONDARYVOLTAGENEG8_0
INTEGER*4 PRIMARYPSVOLTAGEPOS12
INTEGER*4 COMMANDSACCEPTEDCOUNTER
INTEGER*4 COMMANDSREJECTEDCOUNTER
INTEGER*4 PROCESSORBIT
INTEGER*4 BCRTBIT
INTEGER*4 NUMBEROFBADMEMORYBLOCKS
INTEGER*4 SPAREWORD
INTEGER*4 MOSTRECENTEVENTSEQUENCENUMBER
INTEGER*4 MOSTRECENTEVENTMESSAGENUMBER
INTEGER*4 MOSTRECENTEVENTSEQUENCENUMBER2
INTEGER*4 MOSTRECENTEVENTMESSAGENUMBER2
INTEGER*4 MOSTRECENTEVENTSEQUENCENUMBER3
INTEGER*4 MOSTRECENTEVENTMESSAGENUMBER3
INTEGER*4 SAFEHOLDSTATUS

END STRUCTURE

STRUCTURE /L1AVIRS_S2/
  RECORD /SCANTIME/ ScanTime
  CHARACTER ephemerisUsed(10)
  REAL*4 Latitude
  REAL*4 Longitude
  RECORD /PRIMARYHEADER/ primaryHeader
  RECORD /VIRSPACKETHEADER/ virsPacketHeader
  RECORD /L1AVIRS_S2_VIRSHKPACKET/ VIRSHKPACKET
END STRUCTURE

STRUCTURE /VIRSPACKETHEADER/
  CHARACTER SOLARCALDOORDRIVERSELECT
  CHARACTER SCANDRIVELOCK
  CHARACTER SOLARCALDOORCLOSED
  CHARACTER SOLARCALDOOROPEN
  CHARACTER RADCOOLERDOORDRIVERSELECT
  CHARACTER RADCOOLERDOOROUTGAS
  CHARACTER RADCOOLERDOORCLOSED
  CHARACTER RADCOOLERDOOROPEN
  CHARACTER SOLARSPANELSHIELDDOORDRIVERSELECT
  CHARACTER SCANMIRRORSIDE
  CHARACTER SOLARSPANELSHIELDDOORCLOSED
  CHARACTER SOLARSPANELSHIELDDOOROPEN
CHARACTER SOLARPANELSHEILDDSORDRIVERSTATUS
CHARACTER LWIRGAIN
CHARACTER SAFEHOLDSHUTTERBACKUPACTUATOR
CHARACTER SAFEHOLDSHUTTEROPEN
CHARACTER SCANDRIVECIRCUISELECT
CHARACTER PRIMARYSCANDRIVEPOWER
CHARACTER PRIMARY10WOPERATIONALHEATER
CHARACTER PRIMARY5WOPERATIONALHEATER
CHARACTER INFORMATIONEVENTPRESEN'T
CHARACTER REDUNDANTSCANDRIVEPOWER
CHARACTER REDUNDANT10WOPERATIONALHEATER
CHARACTER REDUNDANT5WOPERATIONALHEATER
CHARACTER WARNINGEVENTPRESEN'T
CHARACTER ACTIVEPROCESSOR
CHARACTER SECONDARYPOWERSUPPLY
CHARACTER PRIMARYPOWERSUPPLY
CHARACTER ERROREVENTPRESEN'T
CHARACTER INNERSTAGEHEATER
CHARACTER OUTERSTAGEHEATER
CHARACTER OUTGASENABLE
CHARACTER SERVOLOW
CHARACTER SERVOHIGH
CHARACTER PRIMOUNTRIGHEATER
CHARACTER RTDSOLARSHIELDDACTIVATE
INTEGER*2 PRIMARYBLACKBODYTEMP
CHARACTER RTDSOLARSHIELDENABLED
CHARACTER PADDING
INTEGER*2 SECONDARYBLACKBODYTEMP
CHARACTER DATAMODE
CHARACTER PRISOLARSHIELDDACTIVATE
CHARACTER RTDSOLARSHIELDDACTIVATE
CHARACTER PROCESSORSELFTESTSTATUS
CHARACTER RESERVED
END STRUCTURE

STRUCTURE /L1AVIRS_S1_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1AVIRS_S1_SOLARCAL/
REAL*8 sunVecX
REAL*8 sunVecY
REAL*8 sunVecZ
REAL*8 sunMag
END STRUCTURE

STRUCTURE /NAVIGATION/
REAL*4 scPos(3)
REAL*4 scVel(3)
REAL*4 scLat
REAL*4 scLon
REAL*4 scAlt
REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /PRIMARYHEADER/
BYTE version
BYTE type
BYTE secHeaderFlag
INTEGER*2 APID
BYTE sequenceFlag
INTEGER*2 packetSequenceCount
INTEGER*2 packetLength
END STRUCTURE

STRUCTURE /L1AVIRS_S1_SCANSTATUS/
CHARACTER dataQuality
CHARACTER missing
CHARACTER modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE virsInstS
BYTE virsMode
CHARACTER virsAbnCon
REAL*8 FractionalGranuleNumber
INTEGER*2 attDetermSource
BYTE TRMMcontMode
BYTE TRMMyawUpdateS
BYTE TRMMqac
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1AVIRS_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(261)
  REAL*4 Longitude(261)
  RECORD /L1AVIRS_S1_SCANSTATUS/ scanStatus
  CHARACTER ephemerisUsed(10)
  RECORD /PRIMARYHEADER/ primaryHeader
  RECORD /NAVIGATION/ navigation
  RECORD /L1AVIRS_S1_SOLARCAL/ solarCal
  RECORD /L1AVIRS_S1_SUNDATA/ sunData
  REAL*4 incidenceAngle(261)
  REAL*4 satAzimuthAngle(261)
  REAL*4 solarZenAngle(261)
  REAL*4 solarAzimuthAngle(261)
  REAL*4 sunGlintAngle(261)
  REAL*4 moonVectorInstFrame(3)
5.4 1BASETMI - TMI unpacked packet data

1BASETMI contains TMI science data from the TMI passive microwave instrument flown on the TRMM satellite. There are 4 swaths. Swath S1 has 10V 10H; Swath S2 has 19V, 19H, 21V, 37V, 37H; Swath S3 has 85V, 85H; Swath S4 has Housekeeping.

The S1 channels are:
10.7 GHz vertically-polarized
10.7 GHz horizontally-polarized

The S2 channels are:
18.7 GHz vertically-polarized
18.7 GHz horizontally-polarized
23.8 GHz vertically-polarized
36.5 GHz vertically-polarized
36.5 GHz horizontally-polarized

The S3 channels are:
85.0 GHz vertically-polarized
85.0 GHz horizontally-polarized

S4 has TMI housekeeping.

Earth observations are taken during a segment of the rotation when TMI is looking in the +x direction of the TRMM satellite. Since the spacecraft turns around every few weeks,
+x may be forward or aft. We define the spacecraft axis $v$, used in the definition of the variable $S\text{Cori}nt$ation, at the center of this segment and the same as the $+x$ direction.

Before Aug 7, 2001 $31.6\text{rpm} \times 1\text{min}/60\text{s} \times 5490\text{s/orbit} = 2891$ scans / orbit.

After Aug 24, 2001 $31.6\text{rpm} \times 1\text{min}/60\text{s} \times 5550\text{s/orbit} = 2923$ scans / orbit.

RELATION BETWEEN THE SWATHS: Swath S2 has the same number of scans and the same number of pixels as Swath S1. Swath S3 has the same number of scans and twice as many pixels as Swath S1. Each S1 scan contains 2 channels sampled 104 times along the scan. Each S2 scan contains 5 channels sampled 104 times along the scan. Each S3 scan contains 2 channels sampled 208 times along the scan.

Dimension definitions:
Figure 90 through Figure 119 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 90: Data Format Structure for 1BASETMI, TMI unpacked packet data
5.4 1BASETMI - TMI unpacked packet data

- **S1_SwathHeader**
  - Metadata
    - Group: nscan1
  - ScanTime: 19 bytes
  - Latitude: 4 bytes
  - Longitude: 4 bytes
  - ScanStatus: 27 bytes
  - EphemerisUsed: 1 byte
  - Navigation: 84 bytes
  - Group: nscan1
  - Calibration: 90 bytes
  - Group: nscan1
  - Cal2: 60 bytes
  - Group: nscan1
  - MoonVectorInstFrame: 4 bytes
  - Array: TMIxyz x nscan1
  - CalCounts: 70 bytes
  - Group: nscan1
  - SunData: 40 bytes
  - Group: nscan1
  - IncidenceAngle: 4 bytes
    - Array: nchannel1 x npixelev1 x nscan1
  - SatAzimuthAngle: 4 bytes
    - Array: npixelev1 x nscan1
  - SolarZenAngle: 4 bytes
    - Array: npixelev1 x nscan1
  - SolarAzimuthAngle: 4 bytes
    - Array: npixelev1 x nscan1
  - SunGlintAngle: 4 bytes
    - Array: npixelev1 x nscan1
  - EarthViewCounts: 2 bytes
    - Array: nchannel1 x npixelev1 x nscan1
  - Tb: 4 bytes
    - Array: nchannel1 x npixelev1 x nscan1
  - Ta: 4 bytes
    - Array: nchannel1 x npixelev1 x nscan1
  - MagneticFieldVector: 4 bytes
    - Array: TMIxyz x nscan1
  - RFIFlag: 2 bytes
    - Array: nfreq1 x npixelev1 x nscan1

Figure 91: Data Format Structure for 1BASETMI, S1
Figure 92: Data Format Structure for 1BASETMI, S2
Figure 93: Data Format Structure for 1BASETMI, S3
Figure 94: Data Format Structure for 1BASETMI, S4

Figure 95: Data Format Structure for 1BASETMI, S1, ScanTime
Figure 96: Data Format Structure for 1BASETMI, S1, scanStatus
5 STANDARD GPM PRODUCTS

Figure 97: Data Format Structure for 1BASETMI, S1, navigation

Figure 98: Data Format Structure for 1BASETMI, S1, calibration
5.4 1BASETMI - TMI unpacked packet data

- **moonIndex**: 2 bytes Array: nchannel1 x nscan1
- **hotLoadThermisterTemp**: 4 bytes Array: ntherm x nchannel1 x nscan1
- **WarmIntrusionToColdViewIndex**: 2 bytes Array: nchannel1 x npixelcs1 x nscan1

**Figure 99: Data Format Structure for 1BASETMI, S1, cal2**

- **hotLoadReading**: 2 bytes Array: npixelht1 x nchannel1 x nscan1
- **coldLoadReading**: 2 bytes Array: npixelcs1 x nchannel1 x nscan1
- **hotLoadThermisterCount**: 2 bytes Array: ntherm x nscan1

**Figure 100: Data Format Structure for 1BASETMI, S1, calCounts**

- **solarBetaAngle**: 4 bytes Array: nscan1
- **phaseFromOrbitMidnight**: 4 bytes Array: nscan1
- **sunEarthSeparation**: 4 bytes Array: nscan1
- **earthAngularRadius**: 4 bytes Array: nscan1
- **phaseOfEclipseExit**: 4 bytes Array: nscan1
- **orbitRate**: 4 bytes Array: nscan1
- **timeSinceEclipseEntry**: 4 bytes Array: nscan1
- **sunVectorInBodyFrame**: 4 bytes Array: 3 x nscan1

**Figure 101: Data Format Structure for 1BASETMI, S1, sunData**

- **Year**: 2 bytes Array: nscan2
- **Month**: 1 byte Array: nscan2
- **DayOfMonth**: 1 byte Array: nscan2
- **Hour**: 1 byte Array: nscan2
- **Second**: 1 byte Array: nscan2
- **MilliSecond**: 2 bytes Array: nscan2
- **DayOfYear**: 2 bytes Array: nscan2
- **SecondOfDay**: 8 bytes Array: nscan2

**Figure 102: Data Format Structure for 1BASETMI, S2, ScanTime**
Figure 103: Data Format Structure for 1BASETMI, S2, scanStatus
5.4 1BASETMI - TMI unpacked packet data

Figure 104: Data Format Structure for 1BASETMI, S2, navigation

Figure 105: Data Format Structure for 1BASETMI, S2, calibration
Figure 106: Data Format Structure for 1BASETMI, S2, cal2

- moonIndex: 2 bytes, Array: nchannel2 x nscan2
- hotLoadThermisterTemp: 4 bytes, Array: ntherm x nchannel2 x nscan2
- WarmIntrusionToColdViewIndex: 2 bytes, Array: nchannel2 x npixelcs2 x nscan2

Figure 107: Data Format Structure for 1BASETMI, S2, calCounts

- hotLoadReading: 2 bytes, Array: npixelht2 x nchannel2 x nscan2
- coldLoadReading: 2 bytes, Array: npixelcs2 x nchannel2 x nscan2
- hotLoadThermisterCount: 2 bytes, Array: ntherm x nscan2

Figure 108: Data Format Structure for 1BASETMI, S2, sunData

- solarBetaAngle: 4 bytes, Array: nscan2
- phaseFromOrbitMidnight: 4 bytes, Array: nscan2
- sunEarthSeparation: 4 bytes, Array: nscan2
- earthAngularRadius: 4 bytes, Array: nscan2
- phaseOfEclipseExit: 4 bytes, Array: nscan2
- orbitRate: 4 bytes, Array: nscan2
- timeSinceEclipseEntry: 4 bytes, Array: nscan2
- sunVectorInBodyFrame: 4 bytes, Array: 3 x nscan2

Figure 109: Data Format Structure for 1BASETMI, S3, ScanTime

- Year: 2 bytes, Array: nscan3
- Month: 1 byte, Array: nscan3
- DayOfMonth: 1 byte, Array: nscan3
- Hour: 1 byte, Array: nscan3
- Minute: 1 byte, Array: nscan3
- Second: 1 byte, Array: nscan3
- MilliSecond: 2 bytes, Array: nscan3
- DayOfYear: 2 bytes, Array: nscan3
- SecondOfDay: 8 bytes, Array: nscan3
Figure 110: Data Format Structure for 1BASETMI, S3, scanStatus
Figure 111: Data Format Structure for 1BASETMI, S3, navigation

Figure 112: Data Format Structure for 1BASETMI, S3, calibration
5.4 1BASETMI - TMI unpacked packet data

Figure 113: Data Format Structure for 1BASETMI, S3, cal2

- moonIndex 2 bytes Array: nchannel3 x nscan3
- hotLoadThermistorTemp 4 bytes Array: ntherm x nchannel3 x nscan3
- WarmIntrusionToColdViewIndex 2 bytes Array: nchannel3 x npixels3 x nscan3
Figure 114: Data Format Structure for 1BASETMI, S3, calCounts

Figure 115: Data Format Structure for 1BASETMI, S3, sunData

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**S1** (Swath)
Figure 116: Data Format Structure for 1BASETMI, S4, ScanTime

Figure 117: Data Format Structure for 1BASETMI, S4, primaryHeader
Figure 118: Data Format Structure for 1BASETMI, TMIHKPACKET
5.4 1BASETMI - TMI unpacked packet data

continued from last figure

Figure 119: Data Format Structure for 1BASETMI, S4, TMIHKPACKET

S1_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2
data products. See Metadata for GPM Products for details.

ScanTime (Group in S1)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousands of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime.sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixelev1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixelev1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S1)

dataQuality (1-byte char, array size: nscan1):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not normal</td>
</tr>
<tr>
<td>7</td>
<td>Q&amp;C errors associated with this scan</td>
</tr>
</tbody>
</table>
**missing** (1-byte char, array size: nscan1):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte char, array size: nscan1):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality.

modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is \(2^i\)). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation is not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine tmiIsStatus</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan1):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is \(2^i\)).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.
Bit Meaning if bit = 1
0 Latitude limit exceeded for viewed pixel locations
1 Negative scan time, invalid input
2 Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan1):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

Bit Meaning if bit = 1
0 Ephemeris Gap Interpolated
1 Attitude Gap Interpolated
2 Attitude jump/discontinuity
3 Attitude out of range
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)
SCorientation (2-byte integer, array size: nscan1):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8002</td>
<td>Yaw turn in progress</td>
</tr>
<tr>
<td>-8003</td>
<td>Deep Space Calibration in progress</td>
</tr>
<tr>
<td>-8004</td>
<td>Non-nominal pointing other than above</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan1):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal ACS mode (4) for mission science</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal ACS mode</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan1):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>
targetSelectionMidScan (1-byte integer, array size: nscan1):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

tmiIsStatus (1-byte char, array size: nscan1):
Status of the instrument from Housekeeping packets. Bit 0 is the most significant bit (I.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(8-i) - 1).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>01</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>02</td>
<td>Spare command 1 Status</td>
</tr>
<tr>
<td>03</td>
<td>Spare command 2 Status</td>
</tr>
<tr>
<td>04</td>
<td>1 Hz Clock Select (1=A, 0=B)</td>
</tr>
<tr>
<td>05</td>
<td>Spare</td>
</tr>
<tr>
<td>06</td>
<td>Spare Command 4 Status</td>
</tr>
<tr>
<td>07</td>
<td>Spare Command 5 Status</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

attDetermSource (2-byte integer, array size: nscan1):
Attitude determination source.
A flag explaining how the attitude value was calculated.
Improved estimates make use of ground processing of PR science-instrument-measured roll values, Gyroscope data, and Sun Sensor 1 data. Earlier products (TRMM V7 and before) used the onboard attitudes with various corrections. Values were determined for each granule based on the data available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430 and higher</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
</tbody>
</table>
5.4 1BASETMI - TMI unpacked packet data

421 Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)
413 Reduced accuracy, sun data not available (affecting pitch)
411 Reduced accuracy, PR roll and sun sensor not available
300-399 Reduced accuracy due to various special conditions
200-299 Fallback to using the onboard attitude estimates with TRMM V7 corrections
-91 Spacecraft in safehold mode, no science data
-99 No data due to telemetry data gap

TRMMcontMode (1-byte integer, array size: nscan1):

The Contingency Mode Flag from telemetry indicates alternate attitude control of the spacecraft. The nominal at-launch Attitude Control System (ACS) for TRMM used Earth horizon sensors for pitch and roll control, and the yaw was updated twice each orbit using the Sun Sensors and propagated using gyro data. However, due to possible problems identified with the Earth Sensor Assembly (ESA) lifetime on-orbit, a contingency ACS mode was developed late in the development cycle. This mode used the Sun Sensors, magnetometers, and gyroscope data. It proved very valuable when the horizon sensors had problems with TRMM moving to the higher operating altitude (from 350 to 402.5 km) to extend the mission lifetime. Thus the contingency mode was used throughout the post-boost period. It was also tested early in the mission on 1998-01-13.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal control of spacecraft used in the pre-boost period</td>
</tr>
<tr>
<td>1</td>
<td>Contingency mode control used in the post-boost period</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

TRMMyawUpdateS (1-byte integer, array size: nscan1):

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor
for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMqac** (1-byte integer, array size: nscan1):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

**ephemerisUsed** (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
- 255  Missing value

**navigation** (Group in S1)

**scPos** (4-byte float, array size: XYZ x nscan1):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
- -9999.9  Missing value

**scVel** (4-byte float, array size: XYZ x nscan1):
The velocity vector ($ms^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
- -9999.9  Missing value

**scLat** (4-byte float, array size: nscan1):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
- -9999.9  Missing value

**scLon** (4-byte float, array size: nscan1):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
- -9999.9  Missing value

**scAlt** (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values
range from 350000 to 500000 m. Special values are defined as:
  -9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
  -9999.9 Missing value

**scAttRollGeoc** (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
  -9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
  -9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
  -9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
  -9999.9 Missing value
**scAttPitchGeod** (4-byte float, array size: nscan1):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan1):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan1):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan1):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan1):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**calibration** (Group in S1)

**hotLoadTemp** (4-byte float, array size: nchannel1 x nscan1):
The mean physical temperature for the temperature sensors attached to the hot load. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**coldSkyTemp** (4-byte float, array size: nchannel1 x nscan1):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**onOrbitNonLinearity** (4-byte float, array size: nchannel1 x nscan1):
The on Orbit Non-Linearity. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCount** (2-byte unsigned integer, array size: nchannel1 x nscan1):
The mean Hot Load Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value
meanColdSkyCount (2-byte unsigned integer, array size: nchannel1 x nscan1):
The mean Cold Sky Count. Values range from 0 to 15. Special values are defined as:
65535  Missing value

gain (4-byte float, array size: LNL x nchannel1 x nscan1):
Automatic gain control. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

offset (4-byte float, array size: LNL x nchannel1 x nscan1):
Offset. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

nonLinearGain (4-byte float, array size: nchannel1 x nscan1):
The nonlinear gain. Special values are defined as:
-9999.9  Missing value

calibrationQCflag (2-byte integer, array size: nscan1):
calibrationQCflag. Values range from 0 to 15. Special values are defined as:
-9999  Missing value

receiverTemp (4-byte float, array size: nchannel1 x nscan1):
The receiver temperature. Special values are defined as:
-9999.9  Missing value

receiverGain (4-byte float, array size: nchannel1 x nscan1):
The receiver gain. Special values are defined as:
-9999.9  Missing value

warmInfusionToColdViewIndex (2-byte unsigned integer, array size: nchannel1 x npixelsc1 x nscan1):
Index flag to determine if a cold view sample is contaminated by certain warmer sources. If the value is 0, the sample is good and the count is used in calibration. If the value is non-zero, the sample is contaminated and excluded in calibration.
0: Good sample
1: Bad sample determined by limit check
2: Bad sample determined by 2D medium filter

Values range from 0 to 2. Special values are defined as:
   65535  Missing value

**moonVectorInstFrame** (4-byte float, array size: TMIxyz x nscan1):
The x, y, z components of the moon vector in the GMI instrument coordinate system.
Values are in counts. Special values are defined as:
   -9999.9  Missing value

**calCounts** (Group in S1)

**hotLoadReading** (2-byte unsigned integer, array size: npixelht1 x nchannel1 x nscan1):
Hot Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

**coldLoadReading** (2-byte unsigned integer, array size: npixelcs1 x nchannel1 x nscan1):
Cold Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

**hotLoadThermisterCount** (2-byte unsigned integer, array size: ntherm x nscan1):
Counts from 11 PRTs in the hot load Values range from 0 to 65534 count. Special values are defined as:
   65535  Missing value

**sunData** (Group in S1)

**solarBetaAngle** (4-byte float, array size: nscan1):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
   -9999.9  Missing value

**phaseFromOrbitMidnight** (4-byte float, array size: nscan1):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
   -9999.9  Missing value
sunEarthSeparation (4-byte float, array size: nscan1):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9  Missing value

earthAngularRadius (4-byte float, array size: nscan1):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9  Missing value

phaseOfEclipseExit (4-byte float, array size: nscan1):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9  Missing value

orbitRate (4-byte float, array size: nscan1):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9  Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan1):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9  Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan1):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9  Missing value

incidenceAngle (4-byte float, array size: nchannel1 x npixelev1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9  Missing value

satAzimuthAngle (4-byte float, array size: npixelev1 x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

solarZenAngle (4-byte float, array size: npixelev1 x nscan1):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9  Missing value
**solarAzimuthAngle** (4-byte float, array size: npixelev1 x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npixelev1 x nscan1):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**earthViewCounts** (2-byte unsigned integer, array size: nchannel1 x npixelev1 x nscan1):
Earth view counts. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**Tb** (4-byte float, array size: nchannel1 x npixelev1 x nscan1):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**Ta** (4-byte float, array size: nchannel1 x npixelev1 x nscan1):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**magneticFieldVector** (4-byte float, array size: TMIxyz x nscan1):
Magnetometer volt reading in TAM (x, y, z) coordinate system. Used to perform along-scan correction of earth view counts. (The TAM (x,y,z) coordinate system is similar to GPM S/C coordinate system but y and z axis are rotated by 180 degrees.) Values range from -500 to 500 V. Special values are defined as:
-9999.9 Missing value

**RFIFlag** (2-byte integer, array size: nfreq1 x npixelev1 x nscan1):
Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: No RFI on earth view samples and all Tb values of this swath are lower than or equal to 320 K.
1: Earth view Tb values from one or more channels of this swath are greater than 320 K.
2: RFI on earth view samples is detected by spectral differential method (10 GHz and 19 GHz channels only).
3: (combination of 1 and 2). Earth view Tb values from one or more channels of this swath are greater than 320 K and RFI is detected by spectral differential method (10 GHz and 19 GHz channels only)
-9999: Missing

S2 (Swath)

S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value
DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of
the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixelev2 x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixelev2 x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S2)

dataQuality (1-byte char, array size: nscan2):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not normal</td>
</tr>
<tr>
<td>7</td>
<td>QAC errors associated with this scan</td>
</tr>
</tbody>
</table>

missing (1-byte char, array size: nscan2):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
</tbody>
</table>
modeStatus (1-byte char, array size: nscan2):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation is not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine tmiIsStatus</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nscan2):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
</tbody>
</table>
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan2):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan2):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
</tbody>
</table>
180  -X forward (yaw 180)
-8002 Yaw turn in progress
-8003 Deep Space Calibration in progress
-8004 Non-nominal pointing other than above
-9999 Missing

pointingStatus (2-byte integer, array size: nscan2):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is
good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit
in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal ACS mode (4) for mission science</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal ACS mode</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan2):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan2):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control
System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>
**tmIsStatus** (1-byte char, array size: nscan2):
Status of the instrument from Housekeeping packets. Bit 0 is the most significant bit (I.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^{8-i} - 1$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>01</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>02</td>
<td>Spare command 1 Status</td>
</tr>
<tr>
<td>03</td>
<td>Spare command 2 Status</td>
</tr>
<tr>
<td>04</td>
<td>1 Hz Clock Select (1=A, 0=B)</td>
</tr>
<tr>
<td>05</td>
<td>Spare</td>
</tr>
<tr>
<td>06</td>
<td>Spare Command 4 Status</td>
</tr>
<tr>
<td>07</td>
<td>Spare Command 5 Status</td>
</tr>
</tbody>
</table>

**FractionalGranuleNumber** (8-byte float, array size: nscan2):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**attDetermSource** (2-byte integer, array size: nscan2):
Attitude determination source. A flag explaining how the attitude value was calculated.

Improved estimates make use of ground processing of PR science-instrument-measured roll values, Gyroscope data, and Sun Sensor 1 data. Earlier products (TRMM V7 and before) used the onboard attitudes with various corrections. Values were determined for each granule based on the data available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
<tr>
<td>421</td>
<td>Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)</td>
</tr>
<tr>
<td>413</td>
<td>Reduced accuracy, sun data not available (affecting pitch)</td>
</tr>
<tr>
<td>411</td>
<td>Reduced accuracy, PR roll and sun sensor not available</td>
</tr>
<tr>
<td>300-399</td>
<td>Reduced accuracy due to various special conditions</td>
</tr>
<tr>
<td>200-299</td>
<td>Fallback to using the onboard attitude estimates with TRMM V7 corrections</td>
</tr>
<tr>
<td>-91</td>
<td>Spacecraft in safehold mode, no science data</td>
</tr>
<tr>
<td>-99</td>
<td>No data due to telemetry data gap</td>
</tr>
</tbody>
</table>
TRMMcontMode (1-byte integer, array size: nscan2):

The Contingency Mode Flag from telemetry indicates alternate attitude control of the spacecraft. The nominal at-launch Attitude Control System (ACS) for TRMM used Earth horizon sensors for pitch and roll control, and the yaw was updated twice each orbit using the Sun Sensors and propagated using gyro data. However, due to possible problems identified with the Earth Sensor Assembly (ESA) lifetime on-orbit, a contingency ACS mode was developed late in the development cycle. This mode used the Sun Sensors, magnetometers, and gyroscope data. It proved very valuable when the horizon sensors had problems with TRMM moving to the higher operating altitude (from 350 to 402.5 km) to extend the mission lifetime. Thus the contingency mode was used throughout the post-boost period. It was also tested early in the mission on 1998-01-13.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal control of spacecraft</td>
</tr>
<tr>
<td></td>
<td>used in the pre-boost period</td>
</tr>
<tr>
<td>1</td>
<td>Contingency mode control</td>
</tr>
<tr>
<td></td>
<td>used in the post-boost period</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

TRMMYawUpdateS (1-byte integer, array size: nscan2):

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>
TRMMqac (1-byte integer, array size: nscan2):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

ephemerisUsed (1-byte char, array size: dim10 x nscan2):
The ephemeris source used to geolocate the swath. Special values are defined as:
  255 Missing value

navigation (Group in S2)

scPos (4-byte float, array size: XYZ x nscan2):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
  -9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan2):
The velocity vector \( ms^{-1} \) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
  -9999.9 Missing value

scLat (4-byte float, array size: nscan2):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
  -9999.9 Missing value

scLon (4-byte float, array size: nscan2):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
  -9999.9 Missing value

scAlt (4-byte float, array size: nscan2):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
  -9999.9 Missing value

dprAlt (4-byte float, array size: nscan2):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
  -9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll
for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
- 9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
- 9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan2):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
- 9999.9 Missing value
timeMidScan (8-byte float, array size: nscan2):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. 
timeMidScan is used as the reference time for the scPos and scVel values. Values range 
from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan2):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 
to 100 s. Special values are defined as:
-9999.9 Missing value

calibration (Group in S2)

hotLoadTemp (4-byte float, array size: nchannel2 x nscan2):
The mean physical temperature for the temperature sensors attached to the hot load. 
Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

coldSkyTemp (4-byte float, array size: nchannel2 x nscan2):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined 
as:
-9999.9 Missing value

onOrbitNonLinearity (4-byte float, array size: nchannel2 x nscan2):
The on Orbit Non-Linearity. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

meanHotLoadCount (2-byte unsigned integer, array size: nchannel2 x nscan2):
The mean Hot Load Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value

meanColdSkyCount (2-byte unsigned integer, array size: nchannel2 x nscan2):
The mean Cold Sky Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value

gain (4-byte float, array size: LNL x nchannel2 x nscan2):
Automatic gain control. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

offset (4-byte float, array size: LNL x nchannel2 x nscan2):
Offset. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

nonLinearGain (4-byte float, array size: nchannel2 x nscan2):
The nonlinear gain. Special values are defined as:
-9999.9 Missing value
**5.4 1BASETMI - TMI unpacked packet data**

**calibrationQCflag** (2-byte integer, array size: nscan2):
- **Values range from 0 to 15. Special values are defined as:**
  - -9999  Missing value

**receiverTemp** (4-byte float, array size: nchannel2 x nscan2):
- The receiver temperature. Special values are defined as:
  - -9999.9  Missing value

**receiverGain** (4-byte float, array size: nchannel2 x nscan2):
- The receiver gain. Special values are defined as:
  - -9999.9  Missing value

**cal2** (Group in S2)

**moonIndex** (2-byte unsigned integer, array size: nchannel2 x nscan2):
- Index determined by the angle between moon vector and cold sample vectors. 0 means angles between moon vector and all cold view vectors are greater than 5 degrees. Non-zero value means the number of cold samples that may be contaminated by moon. Values range from 0 to 100. Special values are defined as:
  - 0  Missing value

**hotLoadThermisterTemp** (4-byte float, array size: ntherm x nchannel2 x nscan2):
- Hot Load Thermister Temperature of 11 PRTs. Values range from 0 to 400 K. Special values are defined as:
  - -9999.9  Missing value

**WarmIntrusionToColdViewIndex** (2-byte unsigned integer, array size: nchannel2 x npixelcs2 x nscan2):
- Index flag to determine if a cold view sample is contaminated by certain warmer sources. If the value is 0, the sample is good and the count is used in calibration. If the value is non-zero, the sample is contaminated and excluded in calibration.
  - 0: Good sample
  - 1: Bad sample determined by limit check
  - 2: Bad sample determined by 2D medium filter

Values range from 0 to 2. Special values are defined as:
  - 65535  Missing value

**moonVectorInstFrame** (4-byte float, array size: TMIXyz x nscan2):
- The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
  - -9999.9  Missing value
**calCounts** (Group in S2)

**hotLoadReading** (2-byte unsigned integer, array size: npixelht2 x nchannel2 x nscan2):
Hot Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
0  Missing value

**coldLoadReading** (2-byte unsigned integer, array size: npixelcs2 x nchannel2 x nscan2):
Cold Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
0  Missing value

**hotLoadThermisterCount** (2-byte unsigned integer, array size: ntherm x nscan2):
Counts from 11 PRTs in the hot load. Values range from 0 to 65534 counts. Special values are defined as:
65535  Missing value

**sunData** (Group in S2)

**solarBetaAngle** (4-byte float, array size: nscan2):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9  Missing value

**phaseFromOrbitMidnight** (4-byte float, array size: nscan2):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9  Missing value

**sunEarthSeparation** (4-byte float, array size: nscan2):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9  Missing value

**earthAngularRadius** (4-byte float, array size: nscan2):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9  Missing value

**phaseOfEclipseExit** (4-byte float, array size: nscan2):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow,
based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nsamp2):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nsamp2):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nsamp2):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value

incidenceAngle (4-byte float, array size: npixelx2 x nsamp2):
The angle at the center of the IFOV between the antenna boresight vector and the zenith vector normal to the Earth Ellipsoid. Also known as Satellite Zenith Angle. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

satAzimuthAngle (4-byte float, array size: npixelx2 x nsamp2):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarZenAngle (4-byte float, array size: npixelx2 x nsamp2):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarAzimuthAngle (4-byte float, array size: npixelx2 x nsamp2):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (4-byte float, array size: npixelx2 x nsamp2):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsiod-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector.
When `sunglintAngle` is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**earthViewCounts** (2-byte unsigned integer, array size: `nchannel2 x npixelev2 x nscan2`):
Earth view counts. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**Tb** (4-byte float, array size: `nchannel2 x npixelev2 x nscan2`):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**Ta** (4-byte float, array size: `nchannel2 x npixelev2 x nscan2`):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**magneticFieldVector** (4-byte float, array size: `TMIxyz x nscan2`):
Magnetometer volt reading in TAM (x, y, z) coordinate system. Used to perform along-scan correction of earth view counts. (The TAM (x,y,z) coordinate system is similar to GPM S/C coordinate system but y and z axis are rotated by 180 degrees.) Values range from -500 to 500 V. Special values are defined as:
-9999.9 Missing value

**RFIFlag** (2-byte integer, array size: `nfreq1 x npixelev2 x nscan2`):

Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: No RFI on earth view samples and all Tb values of this swath are lower than or equal to 320 K.
1: Earth view Tb values from one or more channels of this swath are greater than 320 K.
2: RFI on earth view samples is detected by spectral differential method (10 GHz and 19 GHz channels only).
3: (combination of 1 and 2). Earth view Tb values from one or more channels of this swath are greater than 320 K and RFI is detected by spectral differential method (10 GHz and 19 GHz channels only)
-9999: Missing

**S3** (Swath)
S3_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S3)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan3):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan3):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan3):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan3):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan3):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan3):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan3):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan3):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan3):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixelev3 x nscan3):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixelev3 x nscan3): The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S3)

**dataQuality** (1-byte char, array size: nscan3): A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not normal</td>
</tr>
<tr>
<td>7</td>
<td>QAC errors associated with this scan</td>
</tr>
</tbody>
</table>

**missing** (1-byte char, array size: nscan3): Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte char, array size: nscan3): A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:
Bit Meaning if bit = 1
0  Spare (always 0)
1  SCorientation is not 0 or 180
2  pointingStatus not 0
3  Spare (always 0)
4  Non-routine tmIsStatus
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

ggeoERROR (2-byte integer, array size: nscan3):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0  Latitude limit exceeded for viewed pixel locations
1  Negative scan time, invalid input
2  Error getting spacecraft attitude at scan mid-time
3  Error getting spacecraft ephemeris at scan mid-time
4  Invalid input non-unit ray vector for any pixel
5  Ray misses Earth for any pixel with normal pointing
6  Nadir calculation error for subsatellite position
7  Pixel count with geolocation error over threshold
8  Error in getting spacecraft attitude for any pixel
9  Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

ggeoWARNING (2-byte integer, array size: nscan3):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0 the unsigned integer value is \( 2^i \)):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan3):
The positive angle of the spacecraft vector \( \mathbf{v} \) from the satellite forward direction of motion, measured clockwise facing down. We define \( \mathbf{v} \) in the same direction as the spacecraft axis \( +X \), which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8002</td>
<td>Yaw turn in progress</td>
</tr>
<tr>
<td>-8003</td>
<td>Deep Space Calibration in progress</td>
</tr>
<tr>
<td>-8004</td>
<td>Non-nominal pointing other than above</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan3):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
</table>
acsModeMidScan (1-byte integer, array size: nscan3):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan3):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

tmiIsStatus (1-byte char, array size: nscan3):
Status of the instrument from Housekeeping packets. Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(8-i) - 1).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>01</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>02</td>
<td>Spare command 1 Status</td>
</tr>
<tr>
<td>03</td>
<td>Spare command 2 Status</td>
</tr>
<tr>
<td>04</td>
<td>1 Hz Clock Select (1=A, 0=B)</td>
</tr>
<tr>
<td>05</td>
<td>Spare</td>
</tr>
<tr>
<td>06</td>
<td>Spare Command 4 Status</td>
</tr>
<tr>
<td>07</td>
<td>Spare Command 5 Status</td>
</tr>
</tbody>
</table>
**FractionalGranuleNumber** (8-byte float, array size: nscan3):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
- 9999.9 Missing value

**attDetermSource** (2-byte integer, array size: nscan3):

Attitude determination source.
A flag explaining how the attitude value was calculated.
Improved estimates make use of ground processing of PR
science-instrument-measured roll values, Gyroscope data,
and Sun Sensor 1 data. Earlier products (TRMM V7 and before)
used the onboard attitudes with various corrections.
Values were determined for each granule based on the data
available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430 and higher</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
<tr>
<td>421</td>
<td>Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)</td>
</tr>
<tr>
<td>413</td>
<td>Reduced accuracy, sun data not available (affecting pitch)</td>
</tr>
<tr>
<td>411</td>
<td>Reduced accuracy, PR roll and sun sensor not available</td>
</tr>
<tr>
<td>300-399</td>
<td>Reduced accuracy due to various special conditions</td>
</tr>
<tr>
<td>200-299</td>
<td>Fallback to using the onboard attitude estimates with TRMM V7 corrections</td>
</tr>
<tr>
<td>-91</td>
<td>Spacecraft in safehold mode, no science data</td>
</tr>
<tr>
<td>-99</td>
<td>No data due to telemetry data gap</td>
</tr>
</tbody>
</table>

**TRMMcontMode** (1-byte integer, array size: nscan3):

The Contingency Mode Flag from telemetry indicates
alternate attitude control of the spacecraft.
The nominal at-launch Attitude Control System (ACS)
for TRMM used Earth horizon sensors for pitch and
roll control, and the yaw was updated twice each orbit
using the Sun Sensors and propagated using gyro data.
However, due to possible problems identified with
the Earth Sensor Assembly (ESA) lifetime on-orbit,
a contingency ACS mode was developed late in the
development cycle. This mode used the Sun Sensors,
magnetometers, and gyroscope data. It proved very
valuable when the horizon sensors had problems with TRMM moving to the higher operating altitude (from 350 to 402.5 km) to extend the mission lifetime. Thus the contingency mode was used throughout the post-boost period. It was also tested early in the mission on 1998-01-13.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal control of spacecraft used in the pre-boost period</td>
</tr>
<tr>
<td>1</td>
<td>Contingency mode control used in the post-boost period</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMYawUpdateS** (1-byte integer, array size: nscan3):

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMqac** (1-byte integer, array size: nscan3):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

**ephemerisUsed** (1-byte char, array size: dim10 x nscan3):
The ephemeris source used to geolocate the swath. Special values are defined as:
- 255 Missing value

**navigation** (Group in S3)
scPos (4-byte float, array size: XYZ x nscan3):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan3):
The velocity vector (m s$^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan3):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan3):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan3):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan3):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan3):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
**scAttPitchGeoc** (4-byte float, array size: nscan3):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan3):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan3):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan3):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan3):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan3):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan3):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan3):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value
calibration (Group in S3)

hotLoadTemp (4-byte float, array size: nchannel3 x nscan3):
The mean physical temperature for the temperature sensors attached to the hot load. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

coldSkyTemp (4-byte float, array size: nchannel3 x nscan3):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

onOrbitNonLinearity (4-byte float, array size: nchannel3 x nscan3):
The on Orbit Non-Linear. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

meanHotLoadCount (2-byte unsigned integer, array size: nchannel3 x nscan3):
The mean Hot Load Count. Values range from 0 to 15. Special values are defined as:
65535   Missing value

meanColdSkyCount (2-byte unsigned integer, array size: nchannel3 x nscan3):
The mean Cold Sky Count. Values range from 0 to 15. Special values are defined as:
65535   Missing value

gain (4-byte float, array size: LNL x nchannel3 x nscan3):
Automatic gain control. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

offset (4-byte float, array size: LNL x nchannel3 x nscan3):
Offset. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

nonLinearGain (4-byte float, array size: nchannel3 x nscan3):
The nonlinear gain. Special values are defined as:
-9999.9  Missing value

calibrationQCflag (2-byte integer, array size: nscan3):
calibrationQCflag. Values range from 0 to 15. Special values are defined as:
-9999  Missing value

receiverTemp (4-byte float, array size: nchannel3 x nscan3):
The receiver temperature. Special values are defined as:
-9999.9  Missing value

receiverGain (4-byte float, array size: nchannel3 x nscan3):
The receiver gain. Special values are defined as:
-9999.9  Missing value
**cal2** (Group in S3)

**moonIndex** (2-byte unsigned integer, array size: nchannel3 x nscan3):
Index determined by the angle between moon vector and cold sample vectors. 0 means angles between moon vector and all cold view vectors are greater than 5 degrees. Non-zero value means the number of cold samples that may be contaminated by moon. Values range from 0 to 100. Special values are defined as:

0  Missing value

**hotLoadThermisterTemp** (4-byte float, array size: ntherm x nchannel3 x nscan3):
Hot Load Thermister Temperature of 11 PRTs. Values range from 0 to 400 K. Special values are defined as:

-9999.9  Missing value

**WarmIntrusionToColdViewIndex** (2-byte unsigned integer, array size: nchannel3 x npixelcs3 x nscan3):
Index flag to determine if a cold view sample is contaminated by certain warmer sources. If the value is 0, the sample is good and the count is used in calibration. If the value is non-zero, the sample is contaminated and excluded in calibration.

0:  Good sample
1:  Bad sample determined by limit check
2:  Bad sample determined by 2D medium filter

Values range from 0 to 2. Special values are defined as:

65535  Missing value

**moonVectorInstFrame** (4-byte float, array size: TMIxyz x nscan3):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:

-9999.9  Missing value

**calCounts** (Group in S3)

**hotLoadReading** (2-byte unsigned integer, array size: npixelht3 x nchannel3 x nscan3):
Hot Load Reading. Values range from 0 to 65535 counts. Special values are defined as:

0  Missing value

**coldLoadReading** (2-byte unsigned integer, array size: npixelcs3 x nchannel3 x nscan3):
Cold Load Reading. Values range from 0 to 65535 counts. Special values are defined as:

0  Missing value
hotLoadThermisterCount (2-byte unsigned integer, array size: ntherm x nscan3):
Counts from 11 PRTs in the hot load. Values range from 0 to 65534 count. Special values are defined as:
65535 Missing value

sunData (Group in S3)

solarBetaAngle (4-byte float, array size: nscan3):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan3):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan3):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan3):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan3):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nscan3):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan3):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values
range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan3):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npixelev3 x nscan3):
The angle at the center of the IFOV between the antenna boresight vector and the zenith vector normal to the Earth Ellipsoid. Also known as Satellite Zenith Angle. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npixelev3 x nscan3):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: npixelev3 x nscan3):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: npixelev3 x nscan3):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npixelev3 x nscan3):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**earthViewCounts** (2-byte unsigned integer, array size: nchannel3 x npixelev3 x nscan3):
Earth view counts. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**Tb** (4-byte float, array size: nchannel3 x npixelev3 x nscan3):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined
as:
-9999.9 Missing value

**Ta** (4-byte float, array size: nchannel3 x npixelev3 x nscan3):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**magneticFieldVector** (4-byte float, array size: TMIxyz x nscan3):
Magnetometer volt reading in TAM (x, y, z) coordinate system. Used to perform along-scan correction of earth view counts. (The TAM (x,y,z) coordinate system is similar to GPM S/C coordinate system but y and z axis are rotated by 180 degrees.) Values range from -500 to 500 V. Special values are defined as:
-9999.9 Missing value

**RFIFlag** (2-byte integer, array size: nfreq1 x npixelev3 x nscan3):
Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: No RFI on earth view samples and all Tb values of this swath are lower than or equal to 320 K.
1: Earth view Tb values from one or more channels of this swath are greater than 320 K.
-9999: Missing

**S4** (Swath)

**S4_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S4)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value
Month (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
   -99   Missing value

DayOfMonth (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
   -99   Missing value

Hour (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
   -99   Missing value

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
   -99   Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
   -99   Missing value

MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
   -9999  Missing value

DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
   -9999  Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
   -9999.9  Missing value

ephemerisUsed (1-byte char, array size: dim10 x nscan1):
The ephemeris source used to geolocate the swath. Special values are defined as:
   255   Missing value

Latitude (4-byte float, array size: nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
   -9999.9  Missing value

Longitude (4-byte float, array size: nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9  Missing value
primaryHeader (Group in S4)

version (1-byte integer, array size: nscan4):

type (1-byte integer, array size: nscan4):

secHeaderFlag (1-byte integer, array size: nscan4):

APID (2-byte integer, array size: nscan4):

sequenceFlag (1-byte integer, array size: nscan4):

packetSequenceCount (2-byte integer, array size: nscan4):

packetLength (2-byte unsigned integer, array size: nscan4):

hotLoadTemperature1 (2-byte unsigned integer, array size: nscan2):
Hot Load Thermister Count Values are in count. Special values are defined as:
   65535  Missing value

hotLoadTemperature2 (2-byte unsigned integer, array size: nscan2):
Hot Load Thermister Count Values are in count. Special values are defined as:
   65535  Missing value

hotLoadTemperature3 (2-byte unsigned integer, array size: nscan2):
Hot Load Thermister Count Values are in count. Special values are defined as:
   65535  Missing value

posBridgeVolt (2-byte unsigned integer, array size: nscan2):
Positive Bridge Voltage Count. Values are in count. Special values are defined as:
   65535  Missing value

nearZeroVolt (2-byte unsigned integer, array size: nscan2):
Near zero voltage of hot load bridge reference. Values are in count. Special values are
   65535  Missing value
defined as:

gain (1-byte char, array size: nchannelall x nscan2):
Gain for each channel. Special values are defined as:
   255  Missing value

TMIHKPACKET (Group in S4)
baptaMotorCurrent (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

momentumUnbalance (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

spu28vSecondary (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

spu16vSecondary (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

spu6vSecondary (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

spu14vPostRegulatorOutput (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

spu5vDigPostRegulatorOutput (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

spu5vAnaPostRegulatorOutput (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

spare (2-byte unsigned integer, array size: dim6 x nscan4):
Special values are defined as:
0  Missing value

conditioningReferenceANearFullScale (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

conditioningReferenceANearZeroScale (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

powerSupplyShelfTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value

baptaMotorTemp (2-byte unsigned integer, array size: nscan4):
Special values are defined as:
0  Missing value
baptaForwardBearingTemp (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

aftMountingPlaceTemp (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

spuTemp (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

bceTemp (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

admTemp (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

admDeploymentStatusA (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

receiverShelfTemperature (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

droShelfTemp (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

topRadiatorTemp (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

admDeploymentStatusB (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

spare2 (2-byte unsigned integer, array size: dim2 x nscan4):
  Special values are defined as:
    0  Missing value

conditioningReferenceBNearFullScale (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value

conditioningReferenceBNearZeroScale (2-byte unsigned integer, array size: nscan4):
  Special values are defined as:
    0  Missing value
sparer3 (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value
receiverCmdStatus (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value
spinupCmdStatus (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value
spareCmd1Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value
spareCmd2Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value
clockSelectCmdStatus (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value
spareCmd3Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value
spareCmd4Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value
spareCmd5Status (1-byte char, array size: nscan4):
Special values are defined as:
  0  Missing value

C Structure Header file:

```c
#ifndef _TK_1BASETMI_H_
#define _TK_1BASETMI_H_

#ifndef _L1BASETMI_S4_TMIHKPACKET_
#define _L1BASETMI_S4_TMIHKPACKET_

typedef struct {
    unsigned short baptaMotorCurrent;
    unsigned short momentumUnbalance;
    unsigned short spu28vSecondary;
    unsigned short spu16vSecondary;
```
typedef struct {
    signed char version;
    signed char type;
    signed char secHeaderFlag;
    short APID;
    unsigned short spu6vSecondary;
    unsigned short spu14vPostRegulatorOutput;
    unsigned short spu5vDigPostRegulatorOutput;
    unsigned short spu5vAnaPostRegulatorOutput;
    unsigned short spare[6];
    unsigned short conditioningReferenceANearFullScale;
    unsigned short conditioningReferenceANearZeroScale;
    unsigned short powerSupplyShelfTemp;
    unsigned short baptaMotorTemp;
    unsigned short baptaForwardBearingTemp;
    unsigned short aftMountingPlaceTemp;
    unsigned short spuTemp;
    unsigned short bceTemp;
    unsigned short admTemp;
    unsigned short admDeploymentStatusA;
    unsigned short receiverShelfTemperature;
    unsigned short droShelfTemp;
    unsigned short topRadiatorTemp;
    unsigned short admDeploymentStatusB;
    unsigned short spare2[2];
    unsigned short conditioningReferenceBNearFullScale;
    unsigned short conditioningReferenceBNearZeroScale;
    unsigned char spare3;
    unsigned char receiverCmdStatus;
    unsigned char spinupCmdStatus;
    unsigned char spareCmd1Status;
    unsigned char spareCmd2Status;
    unsigned char clockSelectCmdStatus;
    unsigned char spareCmd3Status;
    unsigned char spareCmd4Status;
    unsigned char spareCmd5Status;
} L1BASETMI_S4_TMIHKPACKET;

#ifdef

#define _PRIMARYHEADER_

typedef struct {
    signed char version;
    signed char type;
    signed char secHeaderFlag;
    short APID;
} L1BASETMI_S4_TMIHKPACKET;

#endif

#ifndef _PRIMARYHEADER_
#define _PRIMARYHEADER_

typedef struct {
    signed char version;
    signed char type;
    signed char secHeaderFlag;
    short APID;
} L1BASETMI_S4_TMIHKPACKET;

#endif
signed char sequenceFlag;
short packetSequenceCount;
unsigned short packetLength;
} PRIMARYHEADER;

#endif

#ifndef _L1BASETMI_S4_
#define _L1BASETMI_S4_

typedef struct {
    SCANTIME ScanTime;
    unsigned char ephemerisUsed[10];
    float Latitude;
    float Longitude;
    PRIMARYHEADER primaryHeader;
    unsigned short hotLoadTemperature1;
    unsigned short hotLoadTemperature2;
    unsigned short hotLoadTemperature3;
    unsigned short posBridgeVolt;
    unsigned short nearZeroVolt;
    unsigned char gain[9];
    L1BASETMI_S4_TMIHKPACKET TMIHKPACKET;
} L1BASETMI_S4;
#endif

#ifndef _L1BASETMI_S3_SUNDATA_
#define _L1BASETMI_S3_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1BASETMI_S3_SUNDATA;
#endif
#ifndef _L1BASETMI_S3_CALCOUNTS_
#define _L1BASETMI_S3_CALCOUNTS_

typedef struct {
    unsigned short hotLoadReading[2][16];
    unsigned short coldLoadReading[2][16];
    unsigned short hotLoadThermisterCount[3];
} L1BASETMI_S3_CALCOUNTS;
#endif

#ifndef _L1BASETMI_S3_CAL2_
#define _L1BASETMI_S3_CAL2_

typedef struct {
    unsigned short moonIndex[2];
    float hotLoadThermisterTemp[2][3];
    unsigned short WarmIntrusionToColdViewIndex[16][2];
} L1BASETMI_S3_CAL2;
#endif

#ifndef _L1BASETMI_S3_CALIBRATION_
#define _L1BASETMI_S3_CALIBRATION_

typedef struct {
    float hotLoadTemp[2];
    float coldSkyTemp[2];
    float onOrbitNonLinearity[2];
    unsigned short meanHotLoadCount[2];
    unsigned short meanColdSkyCount[2];
    float gain[2][2];
    float offset[2][2];
    float nonLinearGain[2];
    short calibrationQCflag;
    float receiverTemp[2];
    float receiverGain[2];
} L1BASETMI_S3_CALIBRATION;
#endif

#ifndef _L1BASETMI_S3_SCANSTATUS_
#define _L1BASETMI_S3_SCANSTATUS_

typedef struct {
    unsigned char dataQuality;
    unsigned char missing;
    unsigned char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    unsigned char tmiIsStatus;
    double FractionalGranuleNumber;
    short attDetermSource;
    signed char TRMMcontMode;
    signed char TRMMyawUpdateS;
    signed char TRMMqac;
} L1BASETMI_S3_SCANSTATUS;

#endif

#ifndef _L1BASETMI_S3_
#define _L1BASETMI_S3_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[208];
    float Longitude[208];
    L1BASETMI_S3SCANSTATUS scanStatus;
    unsigned char ephemerisUsed[10];
    NAVIGATION navigation;
    L1BASETMI_S3_CALIBRATION calibration;
    L1BASETMI_S3_CAL2 cal2;
    float moonVectorInstFrame[3];
    L1BASETMI_S3_CALCOUNTS calCounts;
    L1BASETMI_S3_SUNDATA sunData;
    float incidenceAngle[208];
    float satAzimuthAngle[208];
    float solarZenAngle[208];
    float solarAzimuthAngle[208];
    float sunGlintAngle[208];
    unsigned short earthViewCounts[208][2];
    float Tb[208][2];
}
float Ta[208][2];
float magneticFieldVector[3];
short RFIFlag[208][1];
} L1BASETMI_S3;

#endif

#ifndef _L1BASETMI_S2_SUNDATA_
#define _L1BASETMI_S2_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1BASETMI_S2_SUNDATA;

#endif

#ifndef _L1BASETMI_S2_CALCOUNTS_
#define _L1BASETMI_S2_CALCOUNTS_

typedef struct {
    unsigned short hotLoadReading[5][8];
    unsigned short coldLoadReading[5][8];
    unsigned short hotLoadThermisterCount[3];
} L1BASETMI_S2_CALCOUNTS;

#endif

#ifndef _L1BASETMI_S2_CAL2_
#define _L1BASETMI_S2_CAL2_

typedef struct {
    unsigned short moonIndex[5];
    float hotLoadThermisterTemp[5][3];
    unsigned short WarmIntrusionToColdViewIndex[8][5];
} L1BASETMI_S2_CAL2;
#ifndef _L1BASETMI_S2_CALIBRATION_
#define _L1BASETMI_S2_CALIBRATION_

typedef struct {
    float hotLoadTemp[5];
    float coldSkyTemp[5];
    float onOrbitNonLinearity[5];
    unsigned short meanHotLoadCount[5];
    unsigned short meanColdSkyCount[5];
    float gain[5][2];
    float offset[5][2];
    float nonLinearGain[5];
    short calibrationQCflag;
    float receiverTemp[5];
    float receiverGain[5];
} L1BASETMI_S2_CALIBRATION;

#endif

#ifndef _L1BASETMI_S2_SCANSTATUS_
#define _L1BASETMI_S2_SCANSTATUS_

typedef struct {
    unsigned char dataQuality;
    unsigned char missing;
    unsigned char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    unsigned char tmiIsStatus;
    double FractionalGranuleNumber;
    short attDetermSource;
    signed char TRMMcontMode;
    signed char TRMMyawUpdateS;
    signed char TRMMqac;
} L1BASETMI_S2_SCANSTATUS;

#endif
#ifndef _L1BASETMI_S2_
#define _L1BASETMI_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[104];
    float Longitude[104];
    L1BASETMI_S2_SCANSTATUS scanStatus;
    unsigned char ephemerisUsed[10];
    NAVIGATION navigation;
    L1BASETMI_S2_CALIBRATION calibration;
    L1BASETMI_S2_CAL2 cal2;
    float moonVectorInstFrame[3];
    L1BASETMI_S2_CALCOUNTS calCounts;
    L1BASETMI_S2_SUNDATA sunData;
    float incidenceAngle[104];
    float satAzimuthAngle[104];
    float solarZenAngle[104];
    float solarAzimuthAngle[104];
    float sunGlintAngle[104];
    unsigned short earthViewCounts[104][5];
    float Tb[104][5];
    float Ta[104][5];
    float magneticFieldVector[3];
    short RFIFlag[104][1];
} L1BASETMI_S2;

#endif

#ifndef _L1BASETMI_S1_SUNDATA_
#define _L1BASETMI_S1_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarth Separation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1BASETMI_S1_SUNDATA;

#endif


```c
#define _L1BASETMI_S1_CALCOUNTS_

typedef struct {
    unsigned short hotLoadReading[2][8];
    unsigned short coldLoadReading[2][8];
    unsigned short hotLoadThermisterCount[3];
} L1BASETMI_S1_CALCOUNTS;

#endif

#define _L1BASETMI_S1_CAL2_

typedef struct {
    unsigned short moonIndex[2];
    float hotLoadThermisterTemp[2][3];
    unsigned short WarmIntrusionToColdViewIndex[8][2];
} L1BASETMI_S1_CAL2;

#endif

#define _L1BASETMI_S1_CALIBRATION_

typedef struct {
    float hotLoadTemp[2];
    float coldSkyTemp[2];
    float onOrbitNonLinearity[2];
    unsigned short meanHotLoadCount[2];
    unsigned short meanColdSkyCount[2];
    float gain[2][2];
    float offset[2][2];
    float nonLinearGain[2];
    short calibrationQCflag;
    float receiverTemp[2];
    float receiverGain[2];
} L1BASETMI_S1_CALIBRATION;

#endif
```
ifndef _NAVIGATION_
define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;

endif

ifndef _L1BASETMI_S1_SCANSTATUS_
define _L1BASETMI_S1_SCANSTATUS_

typedef struct {
    unsigned char dataQuality;
    unsigned char missing;
    unsigned char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    unsigned char tmiIsStatus;
    double FractionalGranuleNumber;
    short attDetermSource;
    signed char TRMcontMode;
    signed char TRMMyawUpdateS;
    signed char TRMMqac;
} L1BASETMI_S1_SCANSTATUS;

endif
typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

typedef struct { 
    SCANTIME ScanTime;
    float Latitude[104];
    float Longitude[104];
    L1BASETMI_S1_SCANSTATUS scanStatus;
    unsigned char ephemerisUsed[10];
    NAVIGATION navigation;
    L1BASETMI_S1_CALIBRATION calibration;
    L1BASETMI_S1_CAL2 cal2;
    float moonVectorInstFrame[3];
    L1BASETMI_S1_CALCOUNTS calCounts;
    L1BASETMI_S1_SUNDATA sunData;
    float incidenceAngle[104][2];
    float satAzimuthAngle[104];
    float solarZenAngle[104];
    float solarAzimuthAngle[104];
    float sunGlintAngle[104];
    unsigned short earthViewCounts[104][2];
    float Tb[104][2];
};
float Ta[104][2];
float magneticFieldVector[3];
short RFIFlag[104][1];
} L1BASETMI_S1;
#endif

#ifndef _L1BASETMI_SWATHS_
define _L1BASETMI_SWATHS_

typedef struct {
   L1BASETMI_S1 S1;
   L1BASETMI_S2 S2;
   L1BASETMI_S3 S3;
   L1BASETMI_S4 S4;
} L1BASETMI_SWATHS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L1BASETMI_S4_TMIHKPACKET/
   INTEGER*2 baptaMotorCurrent
   INTEGER*2 momentumUnbalance
   INTEGER*2 spu28vSecondary
   INTEGER*2 spu16vSecondary
   INTEGER*2 spu6vSecondary
   INTEGER*2 spu14vPostRegulatorOutput
   INTEGER*2 spu5vDigPostRegulatorOutput
   INTEGER*2 spu5vAnaPostRegulatorOutput
   INTEGER*2 spare(6)
   INTEGER*2 conditioningReferenceANearFullScale
   INTEGER*2 conditioningReferenceANearZeroScale
   INTEGER*2 powerSupplyShelfTemp
   INTEGER*2 baptaMotorTemp
   INTEGER*2 baptaForwardBearingTemp
   INTEGER*2 aftMountingPlaceTemp
   INTEGER*2 spuTemp
   INTEGER*2 bceTemp
   INTEGER*2 admTemp
   INTEGER*2 admDeploymentStatusA
INTEGER*2 receiverShelfTemperature
INTEGER*2 droShelfTemp
INTEGER*2 topRadiatorTemp
INTEGER*2 admDeploymentStatusB
INTEGER*2 spare2(2)
INTEGER*2 conditioningReferenceBNearFullScale
INTEGER*2 conditioningReferenceBNearZeroScale
CHARACTER spare3
CHARACTER receiverCmdStatus
CHARACTER spinupCmdStatus
CHARACTER spareCmd1Status
CHARACTER spareCmd2Status
CHARACTER clockSelectCmdStatus
CHARACTER spareCmd3Status
CHARACTER spareCmd4Status
CHARACTER spareCmd5Status
END STRUCTURE

STRUCTURE /PRIMARYHEADER/
  BYTE version
  BYTE type
  BYTE secHeaderFlag
  INTEGER*2 APID
  BYTE sequenceFlag
  INTEGER*2 packetSequenceCount
  INTEGER*2 packetLength
END STRUCTURE

STRUCTURE /L1BASETMI_S4/
  RECORD /SCANTIME/ ScanTime
  CHARACTER ephemerisUsed(10)
  REAL*4 Latitude
  REAL*4 Longitude
  RECORD /PRIMARYHEADER/ primaryHeader
  INTEGER*2 hotLoadTemperature1
  INTEGER*2 hotLoadTemperature2
  INTEGER*2 hotLoadTemperature3
  INTEGER*2 posBridgeVolt
  INTEGER*2 nearZeroVolt
  CHARACTER gain(9)
  RECORD /L1BASETMI_S4_TMIHKPACKET/ TMIHKPACKET
END STRUCTURE
STRUCTURE /L1BASETMI_S3_SUNDATA/
    REAL*4 solarBetaAngle
    REAL*4 phaseFromOrbitMidnight
    REAL*4 sunEarthSeparation
    REAL*4 earthAngularRadius
    REAL*4 phaseOfEclipseExit
    REAL*4 orbitRate
    REAL*4 timeSinceEclipseEntry
    REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BASETMI_S3_CALCOUNTS/
    INTEGER*2 hotLoadReading(16,2)
    INTEGER*2 coldLoadReading(16,2)
    INTEGER*2 hotLoadThermisterCount(3)
END STRUCTURE

STRUCTURE /L1BASETMI_S3_CAL2/
    INTEGER*2 moonIndex(2)
    REAL*4 hotLoadThermisterTemp(3,2)
    INTEGER*2 WarmIntrusionToColdViewIndex(2,16)
END STRUCTURE

STRUCTURE /L1BASETMI_S3_CALIBRATION/
    REAL*4 hotLoadTemp(2)
    REAL*4 coldSkyTemp(2)
    REAL*4 onOrbitNonLinearity(2)
    INTEGER*2 meanHotLoadCount(2)
    INTEGER*2 meanColdSkyCount(2)
    REAL*4 gain(2,2)
    REAL*4 offset(2,2)
    REAL*4 nonLinearGain(2)
    INTEGER*2 calibrationQCflag
    REAL*4 receiverTemp(2)
    REAL*4 receiverGain(2)
END STRUCTURE

STRUCTURE /L1BASETMI_S3_SCANSTATUS/
    CHARACTER dataQuality
    CHARACTER missing
    CHARACTER modeStatus
    INTEGER*2 geoError
    INTEGER*2 geoWarning

INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
CHARACTER tmiIsStatus
REAL*8 FractionalGranuleNumber
INTEGER*2 attDetermSource
BYTE TRMMcontMode
BYTE TRMMyawUpdateS
BYTE TRMMqac
END STRUCTURE

STRUCTURE /L1BASETMI_S3/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(208)
  REAL*4 Longitude(208)
  RECORD /L1BASETMI_S3_SCANSTATUS/ scanStatus
  CHARACTER ephemerisUsed(10)
  RECORD /NAVIGATION/ navigation
  RECORD /L1BASETMI_S3_CALIBRATION/ calibration
  RECORD /L1BASETMI_S3_CAL2/ cal2
  REAL*4 moonVectorInstFrame(3)
  RECORD /L1BASETMI_S3_CALCOUNTS/ calCounts
  RECORD /L1BASETMI_S3_SUNDATA/ sunData
  REAL*4 incidenceAngle(208)
  REAL*4 satAzimuthAngle(208)
  REAL*4 solarZenAngle(208)
  REAL*4 solarAzimuthAngle(208)
  REAL*4 sunGlintAngle(208)
  INTEGER*2 earthViewCounts(2,208)
  REAL*4 Tb(2,208)
  REAL*4 Ta(2,208)
  REAL*4 magneticFieldVector(3)
  INTEGER*2 RFIFlag(1,208)
END STRUCTURE

STRUCTURE /L1BASETMI_S2_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
REAL*4 timeSinceEclipseEntry
REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BASETMI_S2_CALCOUNTS/
  INTEGER*2 hotLoadReading(8,5)
  INTEGER*2 coldLoadReading(8,5)
  INTEGER*2 hotLoadThermisterCount(3)
END STRUCTURE

STRUCTURE /L1BASETMI_S2_CAL2/
  INTEGER*2 moonIndex(5)
  REAL*4 hotLoadThermisterTemp(3,5)
  INTEGER*2 WarmIntrusionToColdViewIndex(5,8)
END STRUCTURE

STRUCTURE /L1BASETMI_S2_CALIBRATION/
  REAL*4 hotLoadTemp(5)
  REAL*4 coldSkyTemp(5)
  REAL*4 onOrbitNonLinearity(5)
  INTEGER*2 meanHotLoadCount(5)
  INTEGER*2 meanColdSkyCount(5)
  REAL*4 gain(2,5)
  REAL*4 offset(2,5)
  REAL*4 nonLinearGain(5)
  INTEGER*2 calibrationQCflag
  REAL*4 receiverTemp(5)
  REAL*4 receiverGain(5)
END STRUCTURE

STRUCTURE /L1BASETMI_S2_SCANSTATUS/
  CHARACTER dataQuality
  CHARACTER missing
  CHARACTER modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  CHARACTER tmiIsStatus
  REAL*8 FractionalGranuleNumber
  INTEGER*2 attDetermSource
BYTE TRMMcontMode
BYTE TRMMyawUpdateS
BYTE TRMMqac
END STRUCTURE

STRUCTURE /L1BASETMI_S2/
RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(104)
REAL*4 Longitude(104)
RECORD /L1BASETMI_S2_SCANSTATUS/ scanStatus
CHARACTER ephemerisUsed(10)
RECORD /NAVIGATION/ navigation
RECORD /L1BASETMI_S2_CALIBRATION/ calibration
RECORD /L1BASETMI_S2_CAL2/ cal2
REAL*4 moonVectorInstFrame(3)
RECORD /L1BASETMI_S2_CALCOUNTS/ calCounts
RECORD /L1BASETMI_S2_SUNDATA/ sunData
REAL*4 incidenceAngle(104)
REAL*4 satAzimuthAngle(104)
REAL*4 solarZenAngle(104)
REAL*4 solarAzimuthAngle(104)
REAL*4 sunGlintAngle(104)
INTEGER*2 earthViewCounts(5,104)
REAL*4 Tb(5,104)
REAL*4 Ta(5,104)
REAL*4 magneticFieldVector(3)
INTEGER*2 RFIFlag(1,104)
END STRUCTURE

STRUCTURE /L1BASETMI_S1_SUNDATA/
REAL*4 solarBetaAngle
REAL*4 phaseFromOrbitMidnight
REAL*4 sunEarthSeparation
REAL*4 earthAngularRadius
REAL*4 phaseOfEclipseExit
REAL*4 orbitRate
REAL*4 timeSinceEclipseEntry
REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BASETMI_S1_CALCOUNTS/
INTEGER*2 hotLoadReading(8,2)
INTEGER*2 coldLoadReading(8,2)
INTEGER*2 hotLoadThermisterCount(3)
END STRUCTURE

STRUCTURE /L1BASETMI_S1_CAL2/
INTEGER*2 moonIndex(2)
REAL*4 hotLoadThermisterTemp(3,2)
INTEGER*2 WarmIntrusionToColdViewIndex(2,8)
END STRUCTURE

STRUCTURE /L1BASETMI_S1_CALIBRATION/
REAL*4 hotLoadTemp(2)
REAL*4 coldSkyTemp(2)
REAL*4 onOrbitNonLinearity(2)
INTEGER*2 meanHotLoadCount(2)
INTEGER*2 meanColdSkyCount(2)
REAL*4 gain(2,2)
REAL*4 offset(2,2)
REAL*4 nonLinearGain(2)
INTEGER*2 calibrationQCflag
REAL*4 receiverTemp(2)
REAL*4 receiverGain(2)
END STRUCTURE

STRUCTURE /NAVIGATION/
REAL*4 scPos(3)
REAL*4 scVel(3)
REAL*4 scLat
REAL*4 scLon
REAL*4 scAlt
REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1BASETMI_S1_SCANSTATUS/
CHARACTER dataQuality
5.4  L1BASETMI - TMI unpacked packet data

CHARACTER missing
CHARACTER modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
CHARACTER tmiIsStatus
REAL*8 FractionalGranuleNumber
INTEGER*2 attDetermSource
BYTE TRMMcontMode
BYTE TRMMyawUpdateS
BYTE TRMMqac
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASETMI_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(104)
  REAL*4 Longitude(104)
  RECORD /L1BASETMI_S1_SCANSTATUS/ scanStatus
  CHARACTER ephemerisUsed(10)
  RECORD /NAVIGATION/ navigation
  RECORD /L1BASETMI_S1_CALIBRATION/ calibration
  RECORD /L1BASETMI_S1_CAL2/ cal2
  REAL*4 moonVectorInstFrame(3)
  RECORD /L1BASETMI_S1_CALCOUNTS/ calCounts
  RECORD /L1BASETMI_S1_SUNDATA/ sunData
  REAL*4 incidenceAngle(2,104)
  REAL*4 satAzimuthAngle(104)
  REAL*4 solarZenAngle(104)
REAL*4 solarAzimuthAngle(104)
REAL*4 sunGlintAngle(104)
INTEGER*2 earthViewCounts(2,104)
REAL*4 Tb(2,104)
REAL*4 Ta(2,104)
REAL*4 magneticFieldVector(3)
INTEGER*2 RFIFlag(1,104)
END STRUCTURE

STRUCTURE /L1BASETMI_SWATHS/
  RECORD /L1BASETMI_S1/ S1;
  RECORD /L1BASETMI_S2/ S2;
  RECORD /L1BASETMI_S3/ S3;
  RECORD /L1BASETMI_S4/ S4;
END STRUCTURE

5.5 1BASEGMI - GMI Antenna Temperatures

The GMI BASE Product, 1BASEGMI, "GMI Antenna Temperatures," is written as a multi-Swath Structure. Swath S1 has channels 1-9: 10V 10H 19V 19H 23V 37V 37H 89V 89H. Swath S2 has channels 10-13: 166V 166H 183+/-3V 183+/-8V. Swath S3 is like S1 but full scan. Swath S4 is like S2 but full scan. The following sections describe the structure and contents of the format.

Dimension definitions:
Figure 120: Data Format Structure for 1BASEGMI, GMI Antenna Temperatures

nscan var Number of scans in the granule.
nchan1 9 Number of channels in Swath 1.
nchan2 4 Number of channels in Swath 2.
nfreq1 5 Number of frequencies in Swath 1.
nfreq2 2 Number of frequencies in Swath 2.
npix1 221 Number of pixels in Swath 1.
npix2 221 Number of pixels in Swath 2.
npix3 500 Number of pixels in Swath 3.
npix4 500 Number of pixels in Swath 4.
ncolds1 85 Maximum number of cold samples in Swath 1.
ncolds2 85 Maximum number of cold samples in Swath 2.
hhots1 65 Maximum number of hot samples in Swath 1.
hhots2 65 Maximum number of hot samples in Swath 2.
ntherm 11 Number of hot load thermisters.
LNL 2 Linear and non-linear.
nsamt 4 Number of sample types. The types are: total science GSDR, earth-view, hot load, cold sky.
ntach 32 Number of tachometer readings.
GMIxyz 3 x, y, z components in GMI instrument coordinate system.
nndiode 6 Number of noise diodes.
n7 7 Number seven.

Figure 120 through Figure 152 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 121: Data Format Structure for 1BASEGMI, S1
### 5.5 1BASEGMI - GMI Antenna Temperatures

![Data Format Structure for 1BASEGMI, S2](image)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2_SwathHeader</td>
<td>Metadata</td>
<td></td>
</tr>
<tr>
<td>ScanTime</td>
<td>19 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>Latitude</td>
<td>4 bytes, Array: npix2 x nscan</td>
<td></td>
</tr>
<tr>
<td>Longitude</td>
<td>4 bytes, Array: npix2 x nscan</td>
<td></td>
</tr>
<tr>
<td>scanStatus</td>
<td>22 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>sampleHeader</td>
<td>233 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>NEDTInfo</td>
<td>16 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>navigation</td>
<td>84 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>nav2</td>
<td>4 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>calibration</td>
<td>260 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>cal2</td>
<td>998 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>moonVectorInstFrame</td>
<td>4 bytes, Array: GMIxyz x nscan</td>
<td></td>
</tr>
<tr>
<td>calCounts</td>
<td>2422 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>sunData</td>
<td>40 bytes, Group: nscan</td>
<td></td>
</tr>
<tr>
<td>incidenceAngle</td>
<td>4 bytes, Array: npix2 x nscan</td>
<td></td>
</tr>
<tr>
<td>satAzimuthAngle</td>
<td>4 bytes, Array: npix2 x nscan</td>
<td></td>
</tr>
<tr>
<td>solarZenAngle</td>
<td>4 bytes, Array: npix2 x nscan</td>
<td></td>
</tr>
<tr>
<td>solarAzimuthAngle</td>
<td>4 bytes, Array: npix2 x nscan</td>
<td></td>
</tr>
<tr>
<td>sunGlintAngle</td>
<td>4 bytes, Array: npix2 x nscan</td>
<td></td>
</tr>
<tr>
<td>magneticFieldVector</td>
<td>4 bytes, Array: GMIxyz x nscan</td>
<td></td>
</tr>
<tr>
<td>TAMmagneticFieldVector</td>
<td>4 bytes, Array: GMIxyz x nscan</td>
<td></td>
</tr>
<tr>
<td>earthViewCounts</td>
<td>2 bytes, Array: nchan2 x npix2 x nscan</td>
<td></td>
</tr>
<tr>
<td>Tb</td>
<td>4 bytes, Array: nchan2 x npix2 x nscan</td>
<td></td>
</tr>
<tr>
<td>RFIFlag</td>
<td>2 bytes, Array: nfreq2 x npix2 x nscan</td>
<td></td>
</tr>
</tbody>
</table>

Figure 122: Data Format Structure for 1BASEGMI, S2
<table>
<thead>
<tr>
<th><strong>S3_SwathHeader</strong></th>
<th>Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ScanTime</strong></td>
<td>19 bytes</td>
</tr>
<tr>
<td><strong>Latitude</strong></td>
<td>4 bytes</td>
</tr>
<tr>
<td><strong>Longitude</strong></td>
<td>4 bytes</td>
</tr>
<tr>
<td><strong>scanStatus</strong></td>
<td>22 bytes</td>
</tr>
<tr>
<td><strong>calibration</strong></td>
<td>580 bytes</td>
</tr>
<tr>
<td><strong>incidenceAngle</strong></td>
<td>4 bytes</td>
</tr>
<tr>
<td><strong>Tb</strong></td>
<td>4 bytes</td>
</tr>
<tr>
<td><strong>count</strong></td>
<td>2 bytes</td>
</tr>
</tbody>
</table>

Figure 123: Data Format Structure for 1BASEGMI, S3

<table>
<thead>
<tr>
<th><strong>S4_SwathHeader</strong></th>
<th>Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ScanTime</strong></td>
<td>19 bytes</td>
</tr>
<tr>
<td><strong>Latitude</strong></td>
<td>4 bytes</td>
</tr>
<tr>
<td><strong>Longitude</strong></td>
<td>4 bytes</td>
</tr>
<tr>
<td><strong>scanStatus</strong></td>
<td>22 bytes</td>
</tr>
<tr>
<td><strong>calibration</strong></td>
<td>260 bytes</td>
</tr>
<tr>
<td><strong>incidenceAngle</strong></td>
<td>4 bytes</td>
</tr>
<tr>
<td><strong>Tb</strong></td>
<td>4 bytes</td>
</tr>
<tr>
<td><strong>count</strong></td>
<td>2 bytes</td>
</tr>
</tbody>
</table>

Figure 124: Data Format Structure for 1BASEGMI, S4

<table>
<thead>
<tr>
<th><strong>ScanTime</strong></th>
<th>Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Year</strong></td>
<td>2 bytes</td>
</tr>
<tr>
<td><strong>Month</strong></td>
<td>1 byte</td>
</tr>
<tr>
<td><strong>DayOfMonth</strong></td>
<td>1 byte</td>
</tr>
<tr>
<td><strong>Hour</strong></td>
<td>1 byte</td>
</tr>
<tr>
<td><strong>Minute</strong></td>
<td>1 byte</td>
</tr>
<tr>
<td><strong>Second</strong></td>
<td>1 byte</td>
</tr>
<tr>
<td><strong>MilliSecond</strong></td>
<td>2 bytes</td>
</tr>
<tr>
<td><strong>DayOfYear</strong></td>
<td>2 bytes</td>
</tr>
<tr>
<td><strong>SecondOfDay</strong></td>
<td>8 bytes</td>
</tr>
</tbody>
</table>

Figure 125: Data Format Structure for 1BASEGMI, S1, ScanTime
5.5 1BASEGMI - GMI Antenna Temperatures

Figure 126: Data Format Structure for 1BASEGMI, S1, scanStatus

Figure 127: Data Format Structure for 1BASEGMI, S1, sampleHeader

Figure 128: Data Format Structure for 1BASEGMI, S1, NEDTinfo
Figure 129: Data Format Structure for 1BASEGMI, S1, navigation

Figure 130: Data Format Structure for 1BASEGMI, S1, nav2
Figure 131: Data Format Structure for 1BASEGMI, S1, calibration
Figure 132: Data Format Structure for 1BASEGMI, cal2
5.5 1BASEGMI - GMI Antenna Temperatures

continued from last figure

Figure 133: Data Format Structure for 1BASEGMI, S1, cal2

Figure 134: Data Format Structure for 1BASEGMI, S1, calCounts
Figure 135: Data Format Structure for 1BASEGMI, S1, sunData

Figure 136: Data Format Structure for 1BASEGMI, S2, ScanTime
5.5 1BASEGMI - GMI Antenna Temperatures

![Diagram of scanStatus data format structure]

- `dataQuality`: 1 byte, Array: nscan
- `missing`: 1 byte, Array: nscan
- `modeStatus`: 1 byte, Array: nscan
- `geoError`: 2 bytes, Array: nscan
- `geoWarning`: 2 bytes, Array: nscan
- `SCorientation`: 2 bytes, Array: nscan
- `pointingStatus`: 2 bytes, Array: nscan
- `acsModeMidScan`: 1 byte, Array: nscan
- `targetSelectionMidScan`: 1 byte, Array: nscan
- `operationalMode`: 1 byte, Array: nscan
- `FractionalGranuleNumber`: 8 bytes, Array: nscan

Figure 137: Data Format Structure for 1BASEGMI, S2, scanStatus

![Diagram of sampleHeader data format structure]

- `blanking`: 1 byte, Array: nscan
- `earthViewFirstSample`: 2 bytes, Array: nscan
- `sampleNumber`: 2 bytes, Array: nsamt x nchan2 x nscan
- `tachSeconds`: 4 bytes, Array: ntach x nscan
- `tachSubSeconds`: 2 bytes, Array: ntach x nscan
- `indexPulseSeconds`: 4 bytes, Array: nscan
- `indexPulseSubSeconds`: 2 bytes, Array: nscan

Figure 138: Data Format Structure for 1BASEGMI, S2, sampleHeader

![Diagram of NEDTinfo data format structure]

- `NEDTinfo`: 4 bytes, Array: nchan2 x nscan

Figure 139: Data Format Structure for 1BASEGMI, S2, NEDTinfo
Figure 140: Data Format Structure for 1BASEGMI, S2, navigation

Figure 141: Data Format Structure for 1BASEGMI, S2, nav2
5.5 1BASEGMI - GMI Antenna Temperatures

Figure 142: Data Format Structure for 1BASEGMI, S2, calibration
Figure 143: Data Format Structure for 1BASEGMI, cal2
continued from last figure

![Diagram of data structure for 1BASEGMI, S2, cal2](image)

**Figure 144: Data Format Structure for 1BASEGMI, S2, cal2**

![Diagram of data structure for 1BASEGMI, S2, calCounts](image)

**Figure 145: Data Format Structure for 1BASEGMI, S2, calCounts**
Figure 146: Data Format Structure for 1BASEGMI, S2, sunData

Figure 147: Data Format Structure for 1BASEGMI, S3, ScanTime

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in
5.5 1BASEGMI - GMI Antenna Temperatures

**Figure 148: Data Format Structure for 1BASEGMI, S3, scanStatus**

- `dataQuality` 1 byte Array: nscan
- `missing` 1 byte Array: nscan
- `modeStatus` 1 byte Array: nscan
- `geoError` 2 bytes Array: nscan
- `geoWarning` 2 bytes Array: nscan
- `SCorientation` 2 bytes Array: nscan
- `pointingStatus` 2 bytes Array: nscan
- `acsModeMidScan` 1 byte Array: nscan
- `targetSelectionMidScan` 1 byte Array: nscan
- `operationalMode` 1 byte Array: nscan
- `FractionalGranuleNumber` 8 bytes Array: nscan

**Figure 149: Data Format Structure for 1BASEGMI, S3, calibration**

- `hotLoadTemp` 4 bytes Array: nchan1 x nscan
- `coldSkyTemp` 4 bytes Array: nchan1 x nscan
- `onOrbitNonLinearity` 4 bytes Array: nchan1 x nscan
- `derivedNonLinearity` 4 bytes Array: nchan1 x nscan
- `meanHotLoadCount` 4 bytes Array: nchan1 x nscan
- `meanHotLoadCntnDiode` 4 bytes Array: nchan1 x nscan
- `meanColdSkyCount` 4 bytes Array: nchan1 x nscan
- `meanColdSkyCntnDiode` 4 bytes Array: nchan1 x nscan
- `diodeExcessTemp` 4 bytes Array: nchan1 x nscan
- `gain` 4 bytes Array: LNL x nchan1 x nscan
- `offset` 4 bytes Array: LNL x nchan1 x nscan
- `nonLinearGain` 4 bytes Array: nchan1 x nscan
- `calibrationQCflag` 2 bytes Array: nscan
- `diodeFlag` 2 bytes Array: nscan
- `receiverTemp` 4 bytes Array: nchan1 x nscan
- `receiverGain` 4 bytes Array: nchan1 x nscan
Figure 150: Data Format Structure for 1BASEGMI, S4, ScanTime

Figure 151: Data Format Structure for 1BASEGMI, S4, scanStatus
Figure 152: Data Format Structure for 1BASEGMI, S4, calibration
all data products. See Metadata for GPM Products for details.

**S1 (Swath)**

**S1_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S1)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
5.5 1BASEGMI - GMI Antenna Temperatures

day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npix1 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npix1 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S1)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits = 0, the unsigned integer value is $2^i$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit =
1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other
bits = 0, the unsigned integer value is \(2^i\)). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is \(2^i\)).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
</table>


Nominal pointing in Mission Science Mode
1 GPS point solution stale and PVT ephemeris used
2 GEONS solution stale and GEONS ephemeris used
-8000 Non-nominal mission science orientation
-9999 Missing

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**sampleHeader** (Group in S1)

**blanking** (1-byte integer, array size: nscan):

Value of 0 = Blanking off
Value of 1 = Blanking on

**earthViewFirstSample** (2-byte integer, array size: nscan):
Sample number of the first earth view. Values range from 0 to 512. Special values are defined as:
-9999 Missing value

**sampleNumber** (2-byte integer, array size: nsamt x nchan1 x nscan):
Number of valid samples in scan. Values range from 0 to 512. Special values are defined as:
-9999 Missing value

**tachSeconds** (4-byte unsigned integer, array size: ntach x nscan):
Tachometer seconds. Values are in second. Special values are defined as:
0 Missing value

**tachSubSeconds** (2-byte unsigned integer, array size: ntach x nscan):
Tachometer subseconds. Values range from 0 to 62499 in units of 16 microseconds. The missing value is 65535.

**indexPulseSeconds** (4-byte unsigned integer, array size: nscan):
Index Pulse seconds. Values are in second. Special values are defined as:
0 Missing value

**indexPulseSubSeconds** (2-byte unsigned integer, array size: nscan):
Index Pulse subseconds. Values range from 0 to 62499 in units of 16 microseconds. The missing value is 65535.

**NEDTinfo** (Group in S1)

**NEDTinfo** (4-byte float, array size: nchan1 x nscan):
NEDT (Noise Equivalent Differential Temperature) for each channel.

**navigation** (Group in S1)

**scPos** (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector ($\text{m s}^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity
direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that
pitch and roll will have twice orbital frequency components due to the onboard control
system following the oblate geodetic Earth horizon. Note also that the yaw value will
show an orbital frequency component relative to the Earth fixed ground track due to the
Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees.
Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time.
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values
range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order
of the components in the file is roll, pitch, and yaw. However, the angles are computed
using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll
for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic
Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the
Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft veloc-
ity opposite the orbit normal direction, and the X-axis is approximately in the velocity
direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values
range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values
range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coor-
dinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980.
timeMidScan is used as the reference time for the scPos and scVel values. Values range
from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

nav2 (Group in S1)

SCE_SELECTION (2-byte unsigned integer, array size: nscan):
The current SCE selection setting. Special values are defined as:
0 Missing value

SCE_RATE (2-byte unsigned integer, array size: nscan):
The SMA rotational rate reported by the SCE. To obtain the spin rate in RPM, multiply SCE_RATE by 0.002999106 Values range from 1 to 65535 count. Special values are defined as:
0 Missing value

calibration (Group in S1)

hotLoadTemp (4-byte float, array size: nchan1 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray temperature and averaged over closest 5 scans Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

coldSkyTemp (4-byte float, array size: nchan1 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

onOrbitNonLinearity (4-byte float, array size: nchan1 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

derivedNonLinearity (4-byte float, array size: nchan1 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special
values are defined as:
   -9999.9  Missing value

**meanHotLoadCount** (4-byte float, array size: nchan1 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values
range from 0 to 65535 count. Special values are defined as:
   -9999.9  Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Hot Load Plus Noise Diode counts. Averaged over all samples and
closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
   -9999.9  Missing value

**meanColdSkyCount** (4-byte float, array size: nchan1 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range
from 0 to 65535 count. Special values are defined as:
   -9999.9  Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and
closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
   -9999.9  Missing value

**diodeExcessTemp** (4-byte float, array size: nchan1 x nscan):
The Noise Diode Excess Temperature. Cold and diode coupled Temperature=diodeExcessTemp
+ coldSkyTemp. Hot and diode coupled Temperature=diodeExcessTemp + hotLoadTemp. Values range from 0 to 400 K. Special values are defined as:
   -9999.9  Missing value

**gain** (4-byte float, array size: LNL x nchan1 x nscan):
gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
   -9999.9  Missing value

**offset** (4-byte float, array size: LNL x nchan1 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K).
Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan1 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
   -9999.9  Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
calibrationQCflag. value 0 indicates good calibration. Values range from 0 to 15. Special
values are defined as:
   -9999  Missing value

**diodeFlag** (2-byte integer, array size: nscan):
Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

receiverTemp (4-byte float, array size: nchan1 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

receiverGain (4-byte float, array size: nchan1 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9 Missing value

cal2 (Group in S1)

trayTemperatureCount (2-byte unsigned integer, array size: nscan):
Counts to derive hot load tray temperature. Values range from 0 to 65535 count. Special values are defined as:
65535 Missing value

trayTemperature (4-byte float, array size: nscan):
Derive hot load tray temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

moonIndex (2-byte unsigned integer, array size: nchan1 x nscan):
Index determined by the angle between moon vector and cold sample vectors. 0 means angles between moon vector and all cold view vectors are greater than 5 degrees. Non-zero value means the number of cold samples that may be contaminated by moon. Values range from 0 to 100. Special values are defined as:
0 Missing value

noiseDiodeTemp (4-byte float, array size: nndiode x nscan):
Physical temperature of noise diode. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

TEMP_CALRES_2 (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for PRT temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0 Missing value

TEMP_CALRES_1 (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for PRT temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0 Missing value
RS_CALRES_2 (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
  0  Missing value

RS_CALRES_1 (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
  0  Missing value

BATC_CALRES_2 (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
  0  Missing value

BATC_CALRES_1 (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
  0  Missing value

NDIODE_MODE (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Mode. 0 = On every scan, 1 = On every other scan, 2 = Off. Values range from 0 to 2 count. Special values are defined as:
  65535  Missing value

RSST_NDIODE_ST (2-byte unsigned integer, array size: nscan):
Noise diode state during the scan. 0 = Noise diodes OFF for the scan, 1 = Noise diodes ON for the scan. Values range from 0 to 1 count. Special values are defined as:
  65535  Missing value

NDIODE10GHZNUM (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Start Sample Number, i.e., the sample number where noise diodes are turned on. Values range from 0 to 500 count. Special values are defined as:
  65535  Missing value

hotLoadThermisterTemp (4-byte float, array size: ntherm x nchan1 x nscan):
Hot Load Thermister Temperature of 11 PRTs. Values range from 0 to 400 K. Special values are defined as:
  -9999.9  Missing value

TEMP_CALRES_4 (2-byte unsigned integer, array size: nscan):
Low calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
  0  Missing value

TEMP_CALRES_5 (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
  0  Missing value
TEMP_CALRES_6 (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
  0   Missing value

TEMP_89GHZ_LO (4-byte float, array size: nscan):
89 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
  -9999.9   Missing value

TEMP_166GHZ_LO (4-byte float, array size: nscan):
166 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
  -9999.9   Missing value

TEMP_183GHZ_LO (4-byte float, array size: nscan):
183 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
  -9999.9   Missing value

TEMP_V89GHZMXR (4-byte float, array size: nscan):
89 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
  -9999.9   Missing value

TEMP_H89GHZMXR (4-byte float, array size: nscan):
89 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
  -9999.9   Missing value

TEMP_V166GHZMXR (4-byte float, array size: nscan):
166 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
  -9999.9   Missing value

TEMP_H166GHZMXR (4-byte float, array size: nscan):
166 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
  -9999.9   Missing value

TEMP_183GHZMXR (4-byte float, array size: nscan):
183 GHZ Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
  -9999.9   Missing value

TEMP_RS_MR1 (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 1 Values range from 0 to 500 K. Special values are defined as:
  -9999.9   Missing value
5.5 1BASEGMI - GMI Antenna Temperatures

**TEMP_RS_MR2** (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 2 Values range from 0 to 500 K. Special values are defined as:
- -9999.9 Missing value

**MR_ICA_TEMP** (4-byte float, array size: nscan):
Main Reflector Temperature Read By ICA Values range from 0 to 500 K. Special values are defined as:
- -9999.9 Missing value

**MR_LR_LEFT_TEMP** (4-byte float, array size: nscan):
Main Reflector left Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9 Missing value

**MR_LR_RGHT_TEMP** (4-byte float, array size: nscan):
Main Reflector right Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9 Missing value

**MR_LR_LOWR_TEMP** (4-byte float, array size: nscan):
Main Reflector lower Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9 Missing value

**CSR_TEMP1** (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 1 Values range from 0 to 500 K. Special values are defined as:
- -9999.9 Missing value

**CSR_TEMP2** (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 2 Values range from 0 to 500 K. Special values are defined as:
- -9999.9 Missing value

**onOrbitDiodeExcessTemp** (4-byte float, array size: nchan1 x nscan):
Diode Excess Temperature derived from on orbit trended look-up tables as a function of noise diode temperature from telemetry. Values range from 0 to 400 K. Special values are defined as:
- -9999.9 Missing value

**WarmIntrusionToColdViewIndex** (2-byte unsigned integer, array size: nchan1 x ncolds1 x nscan):
Index flag to determine if a cold view sample is contaminated by certain warmer sources. If the value is 0, the sample is good and the count is used in calibration. If the value is non-zero, the sample is contaminated and excluded in calibration.

0: Good sample
1: Bad sample determined by limit check
2: Bad sample determined by 2D medium filter

Values range from 0 to 2. Special values are defined as:
65535 Missing value

**moonVectorInstFrame** (4-byte float, array size: GMIxyz x nscan):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
-9999.9 Missing value

**calCounts** (Group in S1)

**hotLoadReading** (2-byte unsigned integer, array size: nhots1 x nchan1 x nscan):
Hot Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**coldLoadReading** (2-byte unsigned integer, array size: ncolds1 x nchan1 x nscan):
Cold Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**hotLoadnDiodeReading** (2-byte unsigned integer, array size: nhots1 x nchan1 x nscan):
Hot Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**coldLoadnDiodeReading** (2-byte unsigned integer, array size: ncolds1 x nchan1 x nscan):
Cold Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**hotLoadThermisterCount** (2-byte unsigned integer, array size: ntherm x nscan):
Counts from 11 PRTs in the hot load. Values range from 0 to 65534 count. Special values are defined as:
65535 Missing value

**sunData** (Group in S1)

**solarBetaAngle** (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value
5.5 1BASEGMI - GMI Antenna Temperatures

**phaseFromOrbitMidnight** (4-byte float, array size: nscan):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

**sunEarthSeparation** (4-byte float, array size: nscan):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

**earthAngularRadius** (4-byte float, array size: nscan):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**phaseOfEclipseExit** (4-byte float, array size: nscan):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**orbitRate** (4-byte float, array size: nscan):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npix1 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value
satAzimuthAngle (4-byte float, array size: npix1 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction
to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarZenAngle (4-byte float, array size: npix1 x nscan):
The angle between the local pixel geodetic zenith and the direction to the sun. Values
range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

dozaAzimuthAngle (4-byte float, array size: npix1 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction
to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (4-byte float, array size: npix1 x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off
the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location
on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector
from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s
surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle
is the angular separation between the Reflected Satellite View Vector and the Sun Vector.
When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like)
sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

magneticFieldVector (4-byte float, array size: GMIxyz x nscan):
Magnetometer volt reading in TAM (x, y, z) coordinate system. Used to perform along-
scan correction of earth view counts. (The TAM (x,y,z) coordinate system is similar to
GPM S/C coordinate system but y and z axis are rotated by 180 degrees.) Values range
from -500 to 500 V. Special values are defined as:
-9999.9 Missing value

TAMmagneticFieldVector (4-byte float, array size: GMIxyz x nscan):
Magnetic Field derived from GPM three-axis magnetometer (TAM). Values range from
-1000 to 1000 V. Special values are defined as:
-9999.9 Missing value

earthViewCounts (2-byte unsigned integer, array size: nchan1 x npix1 x nscan):
Earth view counts. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

Tb (4-byte float, array size: nchan1 x npix1 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined
as:
-9999.9 Missing value

RFIFlag (2-byte integer, array size: nfreq1 x npix1 x nscan):
Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: Not affected by RFI.
1: Affected by RFI with X-cal filter.
2: Affected by RFI with RSS filter.
3-7: Spare
-9999: Missing

S2 (Swath)

S2.SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value
**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npix2 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npix2 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S2)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \(i = 1\) and other bits = 0, the unsigned integer value is \(2^i\)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
</tbody>
</table>
5.5 1BASEGMI - GMI Antenna Temperatures

4 Housekeeping (HK) telemetry packet missing
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other
bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SC orientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
</tbody>
</table>
8  Error in getting spacecraft attitude for any pixel
9  Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
</tbody>
</table>
pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
Status of the GMI instrument.
Bit Meaning if bit = 1
0 Receiver status (0=ON, 1=OFF)
1 Spinup Status (0=ON, 1=OFF)

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**sampleHeader** (Group in S2)

**blanking** (1-byte integer, array size: nscan):

Value of 0 = Blanking off
Value of 1 = Blanking on

**earthViewFirstSample** (2-byte integer, array size: nscan):
Sample number of the first earth view. Values range from 0 to 512. Special values are
defined as:
-9999 Missing value

**sampleNumber** (2-byte integer, array size: nsamt x nchan2 x nscan):
Number of valid samples in scan. Values range from 0 to 512. Special values are defined
as:
-9999 Missing value

**tachSeconds** (4-byte unsigned integer, array size: ntach x nscan):
Tachometer seconds. Values are in second. Special values are defined as:
0 Missing value

**tachSubSeconds** (2-byte unsigned integer, array size: ntach x nscan):
Tachometer subseconds. Values range from 0 to 62499 in units of 16 microseconds. The
missing value is 65535.

**indexPulseSeconds** (4-byte unsigned integer, array size: nscan):
Index Pulse seconds. Values are in second. Special values are defined as:
0 Missing value

**indexPulseSubSeconds** (2-byte unsigned integer, array size: nscan):
Index Pulse subseconds. Values range from 0 to 62499 in units of 16 microseconds. The
missing value is 65535.
NEDTinfo (Group in S2)

NEDTinfo (4-byte float, array size: nchan2 x nschan):

NEDT (Noise Equivalent Differential Temperature) for each channel.

navigation (Group in S2)

scPos (4-byte float, array size: XYZ x nschan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nschan):
The velocity vector ($ms^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nschan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nschan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nschan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nschan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nschan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed
using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

`scAttPitchGeoc` (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

`scAttYawGeoc` (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

`scAttRollGeod` (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geocentric Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

`scAttPitchGeod` (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

`scAttYawGeod` (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

`greenHourAng` (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coor-
5.5 1BASEGMI - GMI Antenna Temperatures

Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 1000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

nav2 (Group in S2)

SCE_SELECTION (2-byte unsigned integer, array size: nscan):
The current SCE selection setting. Special values are defined as:
0 Missing value

SCE_RATE (2-byte unsigned integer, array size: nscan):
The SMA rotational rate reported by the SCE. To obtain the spin rate in RPM, multiply SCE_RATE by 0.002999106 Values range from 1 to 65535 count. Special values are defined as:
0 Missing value

calibration (Group in S2)

hotLoadTemp (4-byte float, array size: nchan2 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1, 7, 8, 9, 10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2, 11, 12, 13, 14. The values are corrected by tray temperature and averaged over closest 5 scans Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

coldSkyTemp (4-byte float, array size: nchan2 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

onOrbitNonLinearity (4-byte float, array size: nchan2 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table. Values
range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**derivedNonLinearity** (4-byte float, array size: nchan2 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**meanHotLoadCount** (4-byte float, array size: nchan2 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Hot Load Plus Noise Diode counts. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**meanColdSkyCount** (4-byte float, array size: nchan2 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**diodeExcessTemp** (4-byte float, array size: nchan2 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature = diodeExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature = diodeExcessTemp + hotLoad-Temp. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**gain** (4-byte float, array size: LNL x nchan2 x nscan):
gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9  Missing value

**offset** (4-byte float, array size: LNL x nchan2 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan2 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9  Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
calibrationQCflag. value 0 indicates good calibration. Values range from 0 to 15. Special
values are defined as:
-9999 Missing value

**diodeFlag** (2-byte integer, array size: nscan):

Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

**receiverTemp** (4-byte float, array size: nchan2 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**receiverGain** (4-byte float, array size: nchan2 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9 Missing value

**cal2** (Group in S2)

**trayTemperatureCount** (2-byte unsigned integer, array size: nscan):
Counts to derive hot load tray temperature. Values range from 0 to 65534 count. Special values are defined as:
65535 Missing value

**trayTemperature** (4-byte float, array size: nscan):
Derive hot load tray temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**moonIndex** (2-byte unsigned integer, array size: nchan2 x nscan):
Index determined by the angle between moon vector and cold sample vectors. 0 means angles between moon vector and all cold view vectors are greater than 5 degrees. Non-zero value means the number of cold samples that may be contaminated by moon. Values range from 0 to 100. Special values are defined as:
0 Missing value

**noiseDiodeTemp** (4-byte float, array size: nndiode x nscan):
Physical temperature of noise diode. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**TEMP_CALRES.2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for PRT temperature retrieval. Values range from
0 to 65535 count. Special values are defined as:

```
0  Missing value
```

**TEMP_CALRES.1** (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for PRT temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:

```
0  Missing value
```

**RS_CALRES.2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:

```
0  Missing value
```

**RS_CALRES.1** (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:

```
0  Missing value
```

**BATC_CALRES.2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:

```
0  Missing value
```

**BATC_CALRES.1** (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:

```
0  Missing value
```

**NDIODE_MODE** (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Mode. 0 = On every scan, 1 = On every other scan, 2 = Off. Values range from 0 to 2 count. Special values are defined as:

```
0  Missing value
```

**RSST_NDIODE_ST** (2-byte unsigned integer, array size: nscan):
Noise diode state during the scan. 0 = Noise diodes OFF for the scan, 1 = Noise diodes ON for the scan. Values range from 0 to 1 count. Special values are defined as:

```
0  Missing value
```

**NDIODE10GHZNUM** (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Start Sample Number, i.e., the sample number where noise diodes are turned on. Values range from 0 to 500 count. Special values are defined as:

```
0  Missing value
```

**hotLoadThermisterTemp** (4-byte float, array size: ntherm x nchan2 x nscan):
Hot Load Thermister Temperature of 11 PRTs. Values range from 0 to 400 K. Special values are defined as:

```
-9999.9  Missing value
```
5.5 1BASEGMI - GMI Antenna Temperatures

**TEMP_CALRES_4** (2-byte unsigned integer, array size: nscan):
Low calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
0  Missing value

**TEMP_CALRES_5** (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
0  Missing value

**TEMP_CALRES_6** (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
0  Missing value

**TEMP_89GHZ_LO** (4-byte float, array size: nscan):
89 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

**TEMP_166GHZ_LO** (4-byte float, array size: nscan):
166 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

**TEMP_183GHZ_LO** (4-byte float, array size: nscan):
183 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

**TEMP_V89GHZMXR** (4-byte float, array size: nscan):
89 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

**TEMP_H89GHZMXR** (4-byte float, array size: nscan):
89 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

**TEMP_V166GHZMXR** (4-byte float, array size: nscan):
166 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

**TEMP_H166GHZMXR** (4-byte float, array size: nscan):
166 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value
TEMP_183GHZMXR (4-byte float, array size: nscan):
183 GHZ Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_RS_MR1 (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 1 Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_RS_MR2 (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 2 Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

MR_ICA_TEMP (4-byte float, array size: nscan):
Main Reflector Temperature Read By ICA Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

MR_LR_LEFT_TEMP (4-byte float, array size: nscan):
Main Reflector left Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

MR_LR_RGH_TEMP (4-byte float, array size: nscan):
Main Reflector right Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

MR_LR_LOWR_TEMP (4-byte float, array size: nscan):
Main Reflector lower Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

CSR_TEMP1 (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 1 Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

CSR_TEMP2 (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 2 Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

onOrbitDiodeExcessTemp (4-byte float, array size: nchan2 x nscan):
Diode Excess Temperature derived from on orbit trended look-up tables as a function of noise diode temperature from telemetry. Values range from 0 to 400 K. Special values are defined as:
   -9999.9 Missing value
WarmIntrusionToColdViewIndex (2-byte unsigned integer, array size: nchan2 x ncolds2 x nscan):
Index flag to determine if a cold view sample is contaminated by certain warmer sources. If the value is 0, the sample is good and the count is used in calibration. If the value is non-zero, the sample is contaminated and excluded in calibration.

0: Good sample
1: Bad sample determined by limit check
2: Bad sample determined by 2D medium filter

Values range from 0 to 2. Special values are defined as:
65535 Missing value

moonVectorInstFrame (4-byte float, array size: GMIxyz x nscan):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
-9999.9 Missing value

calCounts (Group in S2)

hotLoadReading (2-byte unsigned integer, array size: nhots2 x nchan2 x nscan):
Hot Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

coldLoadReading (2-byte unsigned integer, array size: ncolds2 x nchan2 x nscan):
Cold Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

hotLoadnDiodeReading (2-byte unsigned integer, array size: nhots2 x nchan2 x nscan):
Hot Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

coldLoadnDiodeReading (2-byte unsigned integer, array size: ncolds2 x nchan2 x nscan):
Cold Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

hotLoadThermisterCount (2-byte unsigned integer, array size: ntherm x nscan):
Counts from 11 PRTs in the hot load Values range from 0 to 65534 count. Special values are defined as:
65535 Missing value
sunData (Group in S2)

solarBetaAngle (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nscan):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special
values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npix2 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npix2 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: npix2 x nscan):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: npix2 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npix2 x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth's surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**magneticFieldVector** (4-byte float, array size: GMIxyz x nscan):
Magnetometer volt reading in TAM (x, y, z) coordinate system. Used to perform along-scan correction of earth view counts. (The TAM (x,y,z) coordinate system is similar to GPM S/C coordinate system but y and z axis are rotated by 180 degrees.) Values range from -500 to 500 V. Special values are defined as:
-9999.9 Missing value

**TAMmagneticFieldVector** (4-byte float, array size: GMIxyz x nscan):
Magnetic Field derived from GPM three-axis magnetometer (TAM). Values range from -1000 to 1000 V. Special values are defined as:
-9999.9 Missing value

**earthViewCounts** (2-byte unsigned integer, array size: nchan2 x npix2 x nscan):
Earth view counts. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value
**Tb** (4-byte float, array size: nchan2 x npix2 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**RFIFlag** (2-byte integer, array size: nfreq2 x npix2 x nscan):

Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: Not affected by RFI.
1: Affected by RFI with X-cal filter.
2: Affected by RFI with RSS filter.
3-7: Spare
-9999: Missing

**S3** (Swath)

**S3_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S3)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nscan):  
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:  
-99 Missing value

Second (1-byte integer, array size: nscan):  
Second of the minute. Values range from 0 to 60 s. Special values are defined as:  
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):  
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:  
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):  
Day of the year. Values range from 1 to 366 days. Special values are defined as:  
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):  
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:  
-9999.9 Missing value

Latitude (4-byte float, array size: npix3 x nscan):  
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:  
-9999.9 Missing value

Longitude (4-byte float, array size: npix3 x nscan):  
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:  
-9999.9 Missing value

scanStatus (Group in S3)

dataQuality (1-byte integer, array size: nscan):  
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>
**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit \(i = 1\) and other bits = 0, the unsigned integer value is \(2^i\)). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorintation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit \(i = 1\) and other bits = 0 the unsigned integer value is \(2^i\)).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
</tbody>
</table>
1. Negative scan time, invalid input
2. Error getting spacecraft attitude at scan mid-time
3. Error getting spacecraft ephemeris at scan mid-time
4. Invalid input non-unit ray vector for any pixel
5. Ray misses Earth for any pixel with normal pointing
6. Nadir calculation error for subsatellite position
7. Pixel count with geolocation error over threshold
8. Error in getting spacecraft attitude for any pixel
9. Error in getting spacecraft ephemeris for any pixel
10. Spare (always 0)
11. Spare (always 0)
12. Spare (always 0)
13. Spare (always 0)
14. Spare (always 0)
15. Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (\(v\)) from the satellite forward direction of mo-
tion, measured clockwise facing down. We define \( v \) in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**acsModeMidScan** (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
</tbody>
</table>
5.5 1BASEGMI - GMI Antenna Temperatures

2  S/C Z axis nadir, -X in flight direction
3  Flight Z axis nadir, -X in flight direction
4  +90 yaw for DPR antenna pattern calibration
5  -90 yaw for DPR antenna pattern calibration
-99  Missing

**operationalMode** (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**calibration** (Group in S3)

**hotLoadTemp** (4-byte float, array size: nchan1 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray temperature and averaged over closest 5 scans. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**coldSkyTemp** (4-byte float, array size: nchan1 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**onOrbitNonLinearity** (4-byte float, array size: nchan1 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated look-up table. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**derivedNonLinearity** (4-byte float, array size: nchan1 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value
**meanHotLoadCount** (4-byte float, array size: nchan1 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Hot Load Plus Noise Diode counts Averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCount** (4-byte float, array size: nchan1 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**diodeExcessTemp** (4-byte float, array size: nchan1 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature=diodeExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature=diodeExcessTemp + hotLoadTemp. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**gain** (4-byte float, array size: LNL x nchan1 x nscan):
gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9 Missing value

**offset** (4-byte float, array size: LNL x nchan1 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan1 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9 Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
calibrationQCflag. value 0 indicates good calibration. Values range from 0 to 15. Special values are defined as:
-9999 Missing value

**diodeFlag** (2-byte integer, array size: nscan):
Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

receiverTemp (4-byte float, array size: nchan1 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

receiverGain (4-byte float, array size: nchan1 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9 Missing value

incidenceAngle (4-byte float, array size: npix3 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Tb (4-byte float, array size: nchan1 x npix3 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

count (2-byte unsigned integer, array size: nchan1 x npix3 x nscan):
Full scan count. Values range from 0 to 65534. Special values are defined as:
65535 Missing value

S4 (Swath)

S4_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S4)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value
DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npix4 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npix4 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S4)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).
5.5 1BASEGMI - GMI Antenna Temperatures

Bit Meaning if bit = 1
0  missing
5  geoError is not zero
6  modeStatus is not zero

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

Bit Meaning if bit = 1
0  Scan is missing
1  Science telemetry packet missing
2  Science telemetry segment within packet missing
3  Science telemetry other missing
4  Housekeeping (HK) telemetry packet missing
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

Bit Meaning if bit = 1
0  Spare (always 0)
1  SCorientation not 0 or 180
2  pointingStatus not 0
3  Spare (always 0)
4  Non-routine operationalMode
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0 Latitude limit exceeded for viewed pixel locations
1 Negative scan time, invalid input
2 Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

Bit Meaning if bit = 1
0 Ephemeris Gap Interpolated
1 Attitude Gap Interpolated
2 Attitude jump/discontinuity
3 Attitude out of range
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector \( \mathbf{v} \) from the satellite forward direction of motion, measured clockwise facing down. We define \( \mathbf{v} \) in the same direction as the spacecraft axis \(+X\), which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**acsModeMidScan** (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.
**Value Meaning**

0  S/C Z axis nadir, +X in flight direction
1  Flight Z axis nadir, +X in flight direction
2  S/C Z axis nadir, -X in flight direction
3  Flight Z axis nadir, -X in flight direction
4  +90 yaw for DPR antenna pattern calibration
5  -90 yaw for DPR antenna pattern calibration
-99  Missing

**operationalMode** (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**calibration** (Group in S4)

**hotLoadTemp** (4-byte float, array size: nchan2 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray temperature and averaged over closest 5 scans Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**coldSkyTemp** (4-byte float, array size: nchan2 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**onOrbitNonLinearity** (4-byte float, array size: nchan2 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value
**derivedNonLinearity** (4-byte float, array size: nchan2 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCount** (4-byte float, array size: nchan2 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Hot Load Plus Noise Diode counts. Averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCount** (4-byte float, array size: nchan2 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**diodeExcessTemp** (4-byte float, array size: nchan2 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature=diodeExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature=diodeExcessTemp + hotLoad-Temp. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**gain** (4-byte float, array size: LNL x nchan2 x nscan):
gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9 Missing value

**offset** (4-byte float, array size: LNL x nchan2 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan2 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9 Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
calibrationQCflag. value 0 indicates good calibration. Values range from 0 to 15. Special values are defined as:
-9999 Missing value
diodeFlag (2-byte integer, array size: nscan):

Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

receiverTemp (4-byte float, array size: nchan2 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

receiverGain (4-byte float, array size: nchan2 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9  Missing value

incidenceAngle (4-byte float, array size: npix4 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9  Missing value

Tb (4-byte float, array size: nchan2 x npix4 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

count (2-byte unsigned integer, array size: nchan2 x npix4 x nscan):
Full scan count. Values range from 0 to 65534. Special values are defined as:
65535  Missing value

C Structure Header file:

```c
#ifndef _TK_1BASEGMI_H_
#define _TK_1BASEGMI_H_

#ifndef _L1BASEGMI_S4_CALIBRATION_
#define _L1BASEGMI_S4_CALIBRATION_

typedef struct {
    float hotLoadTemp[4];
    float coldSkyTemp[4];
    float onOrbitNonLinearity[4];
    float derivedNonLinearity[4];
    float meanHotLoadCount[4];
    float meanHotLoadCntnDiode[4];
    float meanColdSkyCount[4];
    float meanColdSkyCntnDiode[4];
} ...
```
float diodeExcessTemp[4];
float gain[4][2];
float offset[4][2];
float nonLinearGain[4];
short calibrationQCflag;
short diodeFlag;
float receiverTemp[4];
float receiverGain[4];
}
L1BASEGMI_S4_CALIBRATION;

#endif

#endif _L1BASEGMI_S4_SCANSTATUS_
#define _L1BASEGMI_S4_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMI_S4_SCANSTATUS;

#endif

#endif _L1BASEGMI_S4_
#define _L1BASEGMI_S4_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[500];
    float Longitude[500];
    L1BASEGMI_S4_SCANSTATUS scanStatus;
    L1BASEGMI_S4_CALIBRATION calibration;
    float incidenceAngle[500];
    float Tb[500][4];
    unsigned short count[500][4];


} L1BASEGMI_S4;

#endif

#ifndef _L1BASEGMI_S3_CALIBRATION_
define _L1BASEGMI_S3_CALIBRATION_

typedef struct {
    float hotLoadTemp[9];
    float coldSkyTemp[9];
    float onOrbitNonLinearity[9];
    float derivedNonLinearity[9];
    float meanHotLoadCount[9];
    float meanHotLoadCntnDiode[9];
    float meanColdSkyCount[9];
    float meanColdSkyCntnDiode[9];
    float diodeExcessTemp[9];
    float gain[9][2];
    float offset[9][2];
    float nonLinearGain[9];
    short calibrationQCflag;
    short diodeFlag;
    float receiverTemp[9];
    float receiverGain[9];
} L1BASEGMI_S3_CALIBRATION;

#endif

#ifndef _L1BASEGMI_S3_SCANSTATUS_
define _L1BASEGMI_S3_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMI_S3_SCANSTATUS;
typedef struct {
  SCANTIME ScanTime;
  float Latitude[500];
  float Longitude[500];
  L1BASEGMI_S3_SCANSTATUS scanStatus;
  L1BASEGMI_S3_CALIBRATION calibration;
  float incidenceAngle[500];
  float Tb[500][9];
  unsigned short count[500][9];
} L1BASEGMI_S3;

typedef struct {
  float solarBetaAngle;
  float phaseFromOrbitMidnight;
  float sunEarthSeparation;
  float earthAngularRadius;
  float phaseOfEclipseExit;
  float orbitRate;
  float timeSinceEclipseEntry;
  float sunVectorInBodyFrame[3];
} L1BASEGMI_S2_SUNDATA;

typedef struct {
  unsigned short hotLoadReading[4][65];
  unsigned short coldLoadReading[4][85];
  unsigned short hotLoadnDiodeReading[4][65];
} L1BASEGMI_S2_CALCOUNTS;
typedef struct {
    unsigned short trayTemperatureCount;
    float trayTemperature;
    unsigned short moonIndex[4];
    float noiseDiodeTemp[6];
    unsigned short TEMP_CALRES_2;
    unsigned short TEMP_CALRES_1;
    unsigned short RS_CALRES_2;
    unsigned short RS_CALRES_1;
    unsigned short BATC_CALRES_2;
    unsigned short BATC_CALRES_1;
    unsigned short NDIODE_MODE;
    unsigned short RSST_NDIODE_ST;
    unsigned short NDIODE10GHZNUM;
    float hotLoadThermisterTemp[4][11];
    unsigned short TEMP_CALRES_4;
    unsigned short TEMP_CALRES_5;
    unsigned short TEMP_CALRES_6;
    float TEMP_89GHZ_LO;
    float TEMP_166GHZ_LO;
    float TEMP_183GHZ_LO;
    float TEMP_V89GHZMXR;
    float TEMP_H89GHZMXR;
    float TEMP_V166GHZMXR;
    float TEMP_H166GHZMXR;
    float TEMP_183GHZMXR;
    float TEMP_RS_MR1;
    float TEMP_RS_MR2;
    float MR_ICA_TEMP;
    float MR_LR_LEFT_TEMP;
    float MR_LR_RGHT_TEMP;
    float MR_LR_LOWR_TEMP;
    float CSR_TEMP1;
    float CSR_TEMP2;
float onOrbitDiodeExcessTemp[4];
unsigned short WarmIntrusionToColdViewIndex[85][4];
} L1BASEGMI_S2_CAL2;

#endif

#ifndef _L1BASEGMI_S2_CALIBRATION_
define _L1BASEGMI_S2_CALIBRATION_

typedef struct {
    float hotLoadTemp[4];
    float coldSkyTemp[4];
    float onOrbitNonLinearity[4];
    float derivedNonLinearity[4];
    float meanHotLoadCount[4];
    float meanHotLoadCntnDiode[4];
    float meanColdSkyCount[4];
    float meanColdSkyCntnDiode[4];
    float diodeExcessTemp[4];
    float gain[4][2];
    float offset[4][2];
    float nonLinearGain[4];
    short calibrationQCflag;
    short diodeFlag;
    float receiverTemp[4];
    float receiverGain[4];
} L1BASEGMI_S2_CALIBRATION;
#endif

#ifndef _L1BASEGMI_S2_NAV2_
define _L1BASEGMI_S2_NAV2_

typedef struct {
    unsigned short SCE_SELECTION;
    unsigned short SCE_RATE;
} L1BASEGMI_S2_NAV2;
#endif

#ifndef _L1BASEGMI_S2_NEDTINFO_
define _L1BASEGMI_S2_NEDTINFO_


typedef struct {
    float NEDTinfo[4];
} L1BASEGMI_S2_NEDTINFO;

#endif

#ifndef _L1BASEGMI_S2_SAMPLEHEADER_
define _L1BASEGMI_S2_SAMPLEHEADER_

typedef struct {
    signed char blanking;
    short earthViewFirstSample;
    short sampleNumber[4][4];
    unsigned int tachSeconds[32];
    unsigned short tachSubSeconds[32];
    unsigned int indexPulseSeconds;
    unsigned short indexPulseSubSeconds;
} L1BASEGMI_S2_SAMPLEHEADER;

#endif

#ifndef _L1BASEGMI_S2_SCANSTATUS_
define _L1BASEGMI_S2_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMI_S2_SCANSTATUS;

#endif

#ifndef _L1BASEGMI_S2_
define _L1BASEGMI_S2_


typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    L1BASEGMI_S2_SCANSTATUS scanStatus;
    L1BASEGMI_S2_SAMPLEHEADER sampleHeader;
    L1BASEGMI_S2_NEDTINFO NEDTinfo;
    NAVIGATION navigation;
    L1BASEGMI_S2_NAV2 nav2;
    L1BASEGMI_S2_CALIBRATION calibration;
    L1BASEGMI_S2_CAL2 cal2;
    float moonVectorInstFrame[3];
    L1BASEGMI_S2_CALCOUTS calCounts;
    L1BASEGMI_S2_SUNDATA sunData;
    float incidenceAngle[221];
    float satAzimuthAngle[221];
    float solarZenAngle[221];
    float solarAzimuthAngle[221];
    float sunGlintAngle[221];
    float magneticFieldVector[3];
    float TAMmagneticFieldVector[3];
    unsigned short earthViewCounts[221][4];
    float Tb[221][4];
    short RFIFlag[221][2];
} L1BASEGMI_S2;

#endif

#ifndef _L1BASEGMI_S1_SUNDATA_
define _L1BASEGMI_S1_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1BASEGMI_S1_SUNDATA;
#endif
#ifndef _L1BASEGMI_S1_CALCOUNTS_
define _L1BASEGMI_S1_CALCOUNTS_

typedef struct {
    unsigned short hotLoadReading[9][65];
    unsigned short coldLoadReading[9][85];
    unsigned short hotLoadnDiodeReading[9][65];
    unsigned short coldLoadnDiodeReading[9][85];
    unsigned short hotLoadThermisterCount[11];
} L1BASEGMI_S1_CALCOUNTS;
#endif

#ifndef _L1BASEGMI_S1_CAL2_
define _L1BASEGMI_S1_CAL2_

typedef struct {
    unsigned short trayTemperatureCount;
    float trayTemperature;
    unsigned short moonIndex[9];
    float noiseDiodeTemp[6];
    unsigned short TEMP_CALRES_2;
    unsigned short TEMP_CALRES_1;
    unsigned short RS_CALRES_2;
    unsigned short RS_CALRES_1;
    unsigned short BATC_CALRES_2;
    unsigned short BATC_CALRES_1;
    unsigned short NDIODE_MODE;
    unsigned short RSST_NDIODE_ST;
    unsigned short NDIODE10GHZNUM;
    float hotLoadThermisterTemp[9][11];
    unsigned short TEMP_CALRES_4;
    unsigned short TEMP_CALRES_5;
    unsigned short TEMP_CALRES_6;
    float TEMP_89GHZ_LO;
    float TEMP_166GHZ_LO;
    float TEMP_183GHZ_LO;
    float TEMP_V89GHZMXR;
    float TEMP_H89GHZMXR;
    float TEMP_V166GHZMXR;
    float TEMP_H166GHZMXR;
    float TEMP_183GHZMXR;
} L1BASEGMI_S1_CAL2;
#endif
float TEMP_RS_MR1;
float TEMP_RS_MR2;
float MR_ICA_TEMP;
float MR_LR_LEFT_TEMP;
float MR_LR_RIGHT_TEMP;
float MR_LR_LOW_TEMP;
float CSR_TEMP1;
float CSR_TEMP2;
float onOrbitDiodeExcessTemp[9];
unsigned short WarmIntrusionToColdViewIndex[85][9];
} L1BASEGMI_S1_CAL2;
#endif

#ifndef _L1BASEGMI_S1_CALIBRATION_
#define _L1BASEGMI_S1_CALIBRATION_

typedef struct {
  float hotLoadTemp[9];
  float coldSkyTemp[9];
  float onOrbitNonLinearity[9];
  float derivedNonLinearity[9];
  float meanHotLoadCount[9];
  float meanHotLoadCntnDiode[9];
  float meanColdSkyCount[9];
  float meanColdSkyCntnDiode[9];
  float diodeExcessTemp[9];
  float gain[9][2];
  float offset[9][2];
  float nonLinearGain[9];
  short calibrationQCflag;
  short diodeFlag;
  float receiverTemp[9];
  float receiverGain[9];
} L1BASEGMI_S1_CALIBRATION;
#endif

#ifndef _L1BASEGMI_S1_NAV2_
#define _L1BASEGMI_S1_NAV2_

typedef struct {
  unsigned short SCE_SELECTION;
  unsigned short WarmIntrusionToColdViewIndex[85][9];
} L1BASEGMI_S1_NAV2;
#endif
unsigned short SCE_RATE;
} L1BASEGMI_S1_NAV2;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;
#endif

#ifndef _L1BASEGMI_S1_NEDTINFO_
#define _L1BASEGMI_S1_NEDTINFO_

typedef struct {
    float NEDTinfo[9];
} L1BASEGMI_S1_NEDTINFO;
#endif

#ifndef _L1BASEGMI_S1_SAMPLEHEADER_
#define _L1BASEGMI_S1_SAMPLEHEADER_

typedef struct {
    signed char blanking;
    short earthViewFirstSample;
} L1BASEGMI_S1_SAMPLEHEADER;
#endif
short sampleNumber[9][4];
unsigned int tachSeconds[32];
unsigned short tachSubSeconds[32];
unsigned int indexPulseSeconds;
unsigned short indexPulseSubSeconds;
} L1BASEGMI_S1_SAMPLEHEADER;

#endif

#ifndef _L1BASEGMI_S1_SCANSTATUS_
define _L1BASEGMI_S1_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    shortgeoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMI_S1_SCANSTATUS;
#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    L1BASEGMI_S1_SCANSTATUS scanStatus;
    L1BASEGMI_S1_SAMPLEHEADER sampleHeader;
    L1BASEGMI_S1_NEDTINFO NEDTinfo;
    NAVIGATION navigation;
    L1BASEGMI_S1_NAV2 nav2;
    L1BASEGMI_S1_CALIBRATION calibration;
    L1BASEGMI_S1_CAL2 cal2;
    float moonVectorInstFrame[3];
    L1BASEGMI_S1_CALCOUNTS calCounts;
    L1BASEGMI_S1_SUNDATA sunData;
    float incidenceAngle[221];
    float satAzimuthAngle[221];
    float solarZenAngle[221];
    float solarAzimuthAngle[221];
    float sunGlintAngle[221];
    float magneticFieldVector[3];
    float TAMmagneticFieldVector[3];
    unsigned short earthViewCounts[221][9];
    float Tb[221][9];
    short RFIFlag[221][5];
} L1BASEGMI_S1;

// Other definitions...

typedef struct {
    L1BASEGMI_S1 S1;
    L1BASEGMI_S2 S2;
    L1BASEGMI_S3 S3;
    L1BASEGMI_S4 S4;
} L1BASEGMI_SWATHS;
#endif

#endif

**Fortran Structure Header file:**

```fortran
STRUCTURE /L1BASEGMI_S4_CALIBRATION/
  REAL*4 hotLoadTemp(4)
  REAL*4 coldSkyTemp(4)
  REAL*4 onOrbitNonLinearity(4)
  REAL*4 derivedNonLinearity(4)
  REAL*4 meanHotLoadCount(4)
  REAL*4 meanHotLoadCntnDiode(4)
  REAL*4 meanColdSkyCount(4)
  REAL*4 meanColdSkyCntnDiode(4)
  REAL*4 diodeExcessTemp(4)
  REAL*4 gain(2,4)
  REAL*4 offset(2,4)
  REAL*4 nonLinearGain(4)
  INTEGER*2 calibrationQCflag
  INTEGER*2 diodeFlag
  REAL*4 receiverTemp(4)
  REAL*4 receiverGain(4)
END STRUCTURE

STRUCTURE /L1BASEGMI_S4_SCANSTATUS/
  BYTE dataQuality
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1BASEGMI_S4/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(500)
  REAL*4 Longitude(500)
```
RECORD /L1BASEGMI_S4_SCANSTATUS/ scanStatus
RECORD /L1BASEGMI_S4_CALIBRATION/ calibration
REAL*4 incidenceAngle(500)
REAL*4 Tb(4,500)
INTEGER*2 count(4,500)
END STRUCTURE

STRUCTURE /L1BASEGMI_S3_CALIBRATION/
REAL*4 hotLoadTemp(9)
REAL*4 coldSkyTemp(9)
REAL*4 onOrbitNonLinearity(9)
REAL*4 derivedNonLinearity(9)
REAL*4 meanHotLoadCount(9)
REAL*4 meanHotLoadCntnDiode(9)
REAL*4 meanColdSkyCount(9)
REAL*4 meanColdSkyCntnDiode(9)
REAL*4 diodeExcessTemp(9)
REAL*4 gain(2,9)
REAL*4 offset(2,9)
REAL*4 nonLinearGain(9)
INTEGER*2 calibrationQCflag
INTEGER*2 diodeFlag
REAL*4 receiverTemp(9)
REAL*4 receiverGain(9)
END STRUCTURE

STRUCTURE /L1BASEGMI_S3_SCANSTATUS/
BYTE dataQuality
BYTE missing
BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1BASEGMI_S3/
RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(500)
5.5 1BASEGMI - GMI Antenna Temperatures

REAL*4 Longitude(500)
RECORD /L1BASEGMI_S3_SCANSTATUS/ scanStatus
RECORD /L1BASEGMI_S3_CALIBRATION/ calibration
REAL*4 incidenceAngle(500)
REAL*4 Tb(9,500)
INTEGER*2 count(9,500)
END STRUCTURE

STRUCTURE /L1BASEGMI_S2_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
  REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BASEGMI_S2_CALCOUNTS/
  INTEGER*2 hotLoadReading(65,4)
  INTEGER*2 coldLoadReading(85,4)
  INTEGER*2 hotLoadnDiodeReading(65,4)
  INTEGER*2 coldLoadnDiodeReading(85,4)
  INTEGER*2 hotLoadThermisterCount(11)
END STRUCTURE

STRUCTURE /L1BASEGMI_S2_CAL2/
  INTEGER*2 trayTemperatureCount
  REAL*4 trayTemperature
  INTEGER*2 moonIndex(4)
  REAL*4 noiseDiodeTemp(6)
  INTEGER*2 TEMP_CALRES_2
  INTEGER*2 TEMP_CALRES_1
  INTEGER*2 RS_CALRES_2
  INTEGER*2 RS_CALRES_1
  INTEGER*2 BATC_CALRES_2
  INTEGER*2 BATC_CALRES_1
  INTEGER*2 NDIODE_MODE
  INTEGER*2 RSST_NDIODE_ST
  INTEGER*2 NDIODE10GHZNUM
  REAL*4 hotLoadThermisterTemp(11,4)
  INTEGER*2 TEMP_CALRES_4
INTEGER*2 TEMP_CALRES_5
INTEGER*2 TEMP_CALRES_6
REAL*4 TEMP_89GHZ_LO
REAL*4 TEMP_166GHZ_LO
REAL*4 TEMP_183GHZ_LO
REAL*4 TEMP_V89GHZMXR
REAL*4 TEMP_H89GHZMXR
REAL*4 TEMP_V166GHZMXR
REAL*4 TEMP_H166GHZMXR
REAL*4 TEMP_183GHZMXR
REAL*4 TEMP_RS_MR1
REAL*4 TEMP_RS_MR2
REAL*4 MR_ICA_TEMP
REAL*4 MR_LR_LEFT_TEMP
REAL*4 MR_LR_RGHT_TEMP
REAL*4 MR_LR_LOWR_TEMP
REAL*4 CSR_TEMP1
REAL*4 CSR_TEMP2
REAL*4 onOrbitDiodeExcessTemp(4)
INTEGER*2 WarmIntrusionToColdViewIndex(4,85)
END STRUCTURE

STRUCTURE /L1BASEGMI_S2_CALIBRATION/
  REAL*4 hotLoadTemp(4)
  REAL*4 coldSkyTemp(4)
  REAL*4 onOrbitNonLinearity(4)
  REAL*4 derivedNonLinearity(4)
  REAL*4 meanHotLoadCount(4)
  REAL*4 meanHotLoadCntnDiode(4)
  REAL*4 meanColdSkyCount(4)
  REAL*4 meanColdSkyCntnDiode(4)
  REAL*4 diodeExcessTemp(4)
  REAL*4 gain(2,4)
  REAL*4 offset(2,4)
  REAL*4 nonLinearGain(4)
  INTEGER*2 calibrationQCflag
  INTEGER*2 diodeFlag
  REAL*4 receiverTemp(4)
  REAL*4 receiverGain(4)
END STRUCTURE

STRUCTURE /L1BASEGMI_S2_NAV2/
  INTEGER*2 SCE_SELECTION
5.5 1BASEGMI - GMI Antenna Temperatures

INTEGER*2 SCE_RATE
END STRUCTURE

STRUCTURE /L1BASEGMI_S2_NEDTINFO/
   REAL*4 NEDTinfo(4)
END STRUCTURE

STRUCTURE /L1BASEGMI_S2_SAMPLEHEADER/
   BYTE blanking
   INTEGER*2 earthViewFirstSample
   INTEGER*2 sampleNumber(4,4)
   INTEGER*4 tachSeconds(32)
   INTEGER*2 tachSubSeconds(32)
   INTEGER*4 indexPulseSeconds
   INTEGER*2 indexPulseSubSeconds
END STRUCTURE

STRUCTURE /L1BASEGMI_S2_SCANSTATUS/
   BYTE dataQuality
   BYTE missing
   BYTE modeStatus
   INTEGER*2 geoError
   INTEGER*2 geoWarning
   INTEGER*2 SCOrientation
   INTEGER*2 pointingStatus
   BYTE acsModeMidScan
   BYTE targetSelectionMidScan
   BYTE operationalMode
   REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1BASEGMI_S2/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(221)
   REAL*4 Longitude(221)
   RECORD /L1BASEGMI_S2_SCANSTATUS/ scanStatus
   RECORD /L1BASEGMI_S2_SAMPLEHEADER/ sampleHeader
   RECORD /L1BASEGMI_S2_NEDTINFO/ NEDTinfo
   RECORD /NAVIGATION/ navigation
   RECORD /L1BASEGMI_S2_NAV2/ nav2
   RECORD /L1BASEGMI_S2_CALIBRATION/ calibration
   RECORD /L1BASEGMI_S2_CAL2/ cal2
   REAL*4 moonVectorInstFrame(3)
RECORD /L1BASEGMI_S2_CALCOUNTS/ calCounts
RECORD /L1BASEGMI_S2_SUNDATA/ sunData
REAL*4 incidenceAngle(221)
REAL*4 satAzimuthAngle(221)
REAL*4 solarZenAngle(221)
REAL*4 solarAzimuthAngle(221)
REAL*4 sunGlintAngle(221)
REAL*4 magneticFieldVector(3)
REAL*4 TAMmagneticFieldVector(3)
INTEGER*2 earthViewCounts(4,221)
REAL*4 Tb(4,221)
INTEGER*2 RFIFlag(2,221)
END STRUCTURE

STRUCTURE /L1BASEGMI_S1_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
  REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BASEGMI_S1_CALCOUNTS/
  INTEGER*2 hotLoadReading(65,9)
  INTEGER*2 coldLoadReading(85,9)
  INTEGER*2 hotLoadnDiodeReading(65,9)
  INTEGER*2 coldLoadnDiodeReading(85,9)
  INTEGER*2 hotLoadThermisterCount(11)
END STRUCTURE

STRUCTURE /L1BASEGMI_S1_CAL2/
  INTEGER*2 trayTemperatureCount
  REAL*4 trayTemperature
  INTEGER*2 moonIndex(9)
  REAL*4 noiseDiodeTemp(6)
  INTEGER*2 TEMP_CALRES_2
  INTEGER*2 TEMP_CALRES_1
  INTEGER*2 RS_CALRES_2
  INTEGER*2 RS_CALRES_1
  INTEGER*2 BATC_CALRES_2
INTEGER*2 BATC_CALRES_1
INTEGER*2 NDIODE_MODE
INTEGER*2 RSST_NDIODE_ST
INTEGER*2 NDIODE10GHZNUM
REAL*4 hotLoadThermisterTemp(11,9)
INTEGER*2 TEMP_CALRES_4
INTEGER*2 TEMP_CALRES_5
INTEGER*2 TEMP_CALRES_6
REAL*4 TEMP_89GHZ_LO
REAL*4 TEMP_166GHZ_LO
REAL*4 TEMP_183GHZ_LO
REAL*4 TEMP_V89GHZMXR
REAL*4 TEMP_H89GHZMXR
REAL*4 TEMP_V166GHZMXR
REAL*4 TEMP_H166GHZMXR
REAL*4 TEMP_183GHZMXR
REAL*4 TEMP_RS_MR1
REAL*4 TEMP_RS_MR2
REAL*4 MR_ICA_TEMP
REAL*4 MR_LR_LEFT_TEMP
REAL*4 MR_LR_RGHT_TEMP
REAL*4 MR_LR_LOWR_TEMP
REAL*4 CSR_TEMP1
REAL*4 CSR_TEMP2
REAL*4 onOrbitDiodeExcessTemp(9)
INTEGER*2 WarmIntrusionToColdViewIndex(9,85)
END STRUCTURE

STRUCTURE /L1BASEGMI_S1_CALIBRATION/
REAL*4 hotLoadTemp(9)
REAL*4 coldSkyTemp(9)
REAL*4 onOrbitNonLinearity(9)
REAL*4 derivedNonLinearity(9)
REAL*4 meanHotLoadCount(9)
REAL*4 meanHotLoadCntnDiode(9)
REAL*4 meanColdSkyCount(9)
REAL*4 meanColdSkyCntnDiode(9)
REAL*4 diodeExcessTemp(9)
REAL*4 gain(2,9)
REAL*4 offset(2,9)
REAL*4 nonLinearGain(9)
INTEGER*2 calibrationQCflag
INTEGER*2 diodeFlag
REAL*4 receiverTemp(9)
REAL*4 receiverGain(9)
END STRUCTURE

STRUCTURE /L1BASEGMI_S1_NAV2/
  INTEGER*2 SCE_SELECTION
  INTEGER*2 SCE_RATE
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
  REAL*4 scAlt
  REAL*4 dprAlt
  REAL*4 scAttRollGeoc
  REAL*4 scAttPitchGeoc
  REAL*4 scAttYawGeoc
  REAL*4 scAttRollGeod
  REAL*4 scAttPitchGeod
  REAL*4 scAttYawGeod
  REAL*4 greenHourAng
  REAL*8 timeMidScan
  REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1BASEGMI_S1_NEDTINFO/
  REAL*4 NEDTinfo(9)
END STRUCTURE

STRUCTURE /L1BASEGMI_S1_SAMPLEHEADER/
  BYTE blanking
  INTEGER*2 earthViewFirstSample
  INTEGER*2 sampleNumber(4,9)
  INTEGER*4 tachSeconds(32)
  INTEGER*2 tachSubSeconds(32)
  INTEGER*4 indexPulseSeconds
  INTEGER*2 indexPulseSubSeconds
END STRUCTURE

STRUCTURE /L1BASEGMI_S1_SCANSTATUS/
  BYTE dataQuality
BYTE missing
BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /SCANTIME/
INTEGER*2 Year
BYTE Month
BYTE DayOfMonth
BYTE Hour
BYTE Minute
BYTE Second
INTEGER*2 MilliSecond
INTEGER*2 DayOfYear
REAL*8 SecondOfDay

END STRUCTURE

STRUCTURE /L1BASEGMI_S1/
RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(221)
REAL*4 Longitude(221)
RECORD /L1BASEGMI_S1_SCANSTATUS/ scanStatus
RECORD /L1BASEGMI_S1_SAMPLEHEADER/ sampleHeader
RECORD /L1BASEGMI_S1_NEDTINFO/ NEDTinfo
RECORD /NAVIGATION/ navigation
RECORD /L1BASEGMI_S1_NAV2/ nav2
RECORD /L1BASEGMI_S1_CALIBRATION/ calibration
RECORD /L1BASEGMI_S1_CAL2/ cal2
REAL*4 moonVectorInstFrame(3)
RECORD /L1BASEGMI_S1_CALCOUNTS/ calCounts
RECORD /L1BASEGMI_S1_SUNDATA/ sunData
REAL*4 incidenceAngle(221)
REAL*4 satAzimuthAngle(221)
REAL*4 solarZenAngle(221)
REAL*4 solarAzimuthAngle(221)
REAL*4 sunGlintAngle(221)
REAL*4 magneticFieldVector(3)
REAL*4 TAMmagneticFieldVector(3)
INTEGER*2 earthViewCounts(9,221)
REAL*4 Tb(9,221)
INTEGER*2 RFIFlag(5,221)
END STRUCTURE

STRUCTURE /L1BASEGMI_SWATHS/
  RECORD /L1BASEGMI_S1/ S1;
  RECORD /L1BASEGMI_S2/ S2;
  RECORD /L1BASEGMI_S3/ S3;
  RECORD /L1BASEGMI_S4/ S4;
END STRUCTURE

5.6 1BASEGMIRSS - GMI Antenna Temperatures

The GMI BASE Product, 1BASEGMIRSS, "GMI Antenna Temperatures," is written as a multi-Swath Structure. Swath S1 has channels 1-9: 10V 10H 19V 19H 23V 37V 37H 89V 89H. Swath S2 has channels 10-13: 166V 166H 183+/−3V 183+/−8V. 1BASEGMIRSS is like 1BASEGMI but has overlap of 200 scans on each end. The following sections describe the structure and contents of the format.

Dimension definitions:
Figure 153: Data Format Structure for 1BASEGMIRSS, GMI Antenna Temperatures

nscan  var  Number of scans in the granule.
nchan1  9  Number of channels in Swath 1.
nchan2  4  Number of channels in Swath 2.
nfreq1  5  Number of frequencies in Swath 1.
nfreq2  2  Number of frequencies in Swath 2.
npix1  221  Number of pixels in Swath 1.
npix2  221  Number of pixels in Swath 2.
npix3  500  Number of pixels in Swath 3.
npix4  500  Number of pixels in Swath 4.
ncolds1  85  Maximum number of cold samples in Swath 1.
ncolds2  85  Maximum number of cold samples in Swath 2.
hots1  65  Maximum number of hot samples in Swath 1.
hots2  65  Maximum number of hot samples in Swath 2.
therm  11  Number of hot load thermisters.
LNL  2  Linear and non-linear.
nsamt  4  Number of sample types. The types are: total science GSDR, earth-view, hot load, cold sky.
ntach  32  Number of tachometer readings.
GMIxyz  3  x, y, z components in GMI instrument coordinate system.
nndiode  6  Number of noise diodes.
n7  7  Number seven.

Figure 153 through Figure 185 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 154: Data Format Structure for 1BASEGMI, S1
Figure 155: Data Format Structure for 1BASEGMIRSS, S2
Figure 156: Data Format Structure for BASEGMIRSS, S3

Figure 157: Data Format Structure for BASEGMIRSS, S4

Figure 158: Data Format Structure for BASEGMIRSS, S1, ScanTime
Figure 159: Data Format Structure for 1BASEGMIRSS, S1, scanStatus

Figure 160: Data Format Structure for 1BASEGMIRSS, S1, sampleHeader

Figure 161: Data Format Structure for 1BASEGMIRSS, S1, NEDTinfo
Figure 162: Data Format Structure for 1BASEGMRSS, S1, navigation

Figure 163: Data Format Structure for 1BASEGMRSS, S1, nav2
Figure 164: Data Format Structure for 1BASEGMIRSS, S1, calibration
Figure 165: Data Format Structure for 1BASEGMIRSS, cal2
5.6 1BASEGMIRSS - GMI Antenna Temperatures

continued from last figure

Figure 166: Data Format Structure for 1BASEGMIRSS, S1, cal2

Figure 167: Data Format Structure for 1BASEGMIRSS, S1, calCounts
Figure 168: Data Format Structure for 1BASEGMIRSS, S1, sunData

- solarBetaAngle: 4 bytes, Array: nscan
- phaseFromOrbitMidnight: 4 bytes, Array: nscan
- sunEarthSeparation: 4 bytes, Array: nscan
- earthAngularRadius: 4 bytes, Array: nscan
- phaseOfEclipseExit: 4 bytes, Array: nscan
- orbitRate: 4 bytes, Array: nscan
- timeSinceEclipseEntry: 4 bytes, Array: nscan
- sunVectorInBodyFrame: 4 bytes, Array: 3 x nscan

Figure 169: Data Format Structure for 1BASEGMIRSS, S2, ScanTime

- Year: 2 bytes, Array: nscan
- Month: 1 byte, Array: nscan
- DayOfMonth: 1 byte, Array: nscan
- Hour: 1 byte, Array: nscan
- Minute: 1 byte, Array: nscan
- Second: 1 byte, Array: nscan
- MilliSecond: 2 bytes, Array: nscan
- DayOfYear: 2 bytes, Array: nscan
- SecondOfDay: 8 bytes, Array: nscan
5.6 1BASEGMIRSS - GMI Antenna Temperatures

Figure 170: Data Format Structure for 1BASEGMIRSS, S2, scanStatus

Figure 171: Data Format Structure for 1BASEGMIRSS, S2, sampleHeader

Figure 172: Data Format Structure for 1BASEGMIRSS, S2, NEDTinfo
Figure 173: Data Format Structure for 1BASEGMIRSS, S2, navigation

Figure 174: Data Format Structure for 1BASEGMIRSS, S2, nav2
5.6 1BASEGMIRSS - GMI Antenna Temperatures

Figure 175: Data Format Structure for 1BASEGMIRSS, S2, calibration
Figure 176: Data Format Structure for 1BASEGMIRSS, cal2
5.6 1BASEGMIRSS - GMI Antenna Temperatures

continued from last figure

- TEMP_183GHZMXR 4 bytes Array: nscan
- TEMP_RS_MR1 4 bytes Array: nscan
- TEMP_RS_MR2 4 bytes Array: nscan
- MR_JCA_TEMP 4 bytes Array: nscan
- MR_LR_LEFT_TEMP 4 bytes Array: nscan
- MR_LR_RGHT_TEMP 4 bytes Array: nscan
- MR_LR_LOWR_TEMP 4 bytes Array: nscan
- CSR_TEMP1 4 bytes Array: nscan
- CSR_TEMP2 4 bytes Array: nscan
- onOrbitDiodeExcessTemp 4 bytes Array: nchan2 x nscan
- WarmIntrusionToColdViewIndex 2 bytes Array: nchan2 x ncolds2 x nscan

Figure 177: Data Format Structure for 1BASEGMIRSS, S2, cal2

- hotLoadReading 2 bytes Array: nhots2 x nchan2 x nscan
- coldLoadReading 2 bytes Array: ncolds2 x nchan2 x nscan
- hotLoadnDiodeReading 2 bytes Array: nhots2 x nchan2 x nscan
- coldLoadnDiodeReading 2 bytes Array: ncolds2 x nchan2 x nscan
- hotLoadThermisterCount 2 bytes Array: ntherm x nscan

Figure 178: Data Format Structure for 1BASEGMIRSS, S2, calCounts
FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in
Figure 181: Data Format Structure for 1BASEGMIRSS, S3, scanStatus

Figure 182: Data Format Structure for 1BASEGMIRSS, S3, calibration
494  5  STANDARD GPM PRODUCTS

Figure 183: Data Format Structure for 1BASEGMRSS, S4, ScanTime

Figure 184: Data Format Structure for 1BASEGMRSS, S4, scanStatus
Figure 185: Data Format Structure for 1BASEGMIRSS, S4, calibration
all data products. See Metadata for GPM Products for details.

**S1** (Swath)

**S1_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S1)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. `scanTime_sec` is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npix1 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npix1 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S1)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>1</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>2</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

Bit Meaning if bit = 1
0 Spare (always 0)
1 SCorientation not 0 or 180
2 pointingStatus not 0
3 Spare (always 0)
4 Non-routine operationalMode
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)

geoError (2-byte integer, array size: nscan): A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0 Latitude limit exceeded for viewed pixel locations
1 Negative scan time, invalid input
2 Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)
geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
</table>
0 Nominal pointing in Mission Science Mode
1 GPS point solution stale and PVT ephemeris used
2 GEONS solution stale and GEONS ephemeris used
-8000 Non-nominal mission science orientation
-9999 Missing

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**sampleHeader** (Group in S1)

**blanking** (1-byte integer, array size: nscan):

Value of 0 = Blanking off
Value of 1 = Blanking on

**earthViewFirstSample** (2-byte integer, array size: nscan):
Sample number of the first earth view. Values range from 0 to 512. Special values are defined as:
-9999 Missing value

**sampleNumber** (2-byte integer, array size: nsamt x nchan1 x nscan):
Number of valid samples in scan. Values range from 0 to 512. Special values are defined as:
-9999 Missing value

**tachSeconds** (4-byte unsigned integer, array size: ntach x nscan):
Tachometer seconds. Values are in second. Special values are defined as:
0 Missing value

**tachSubSeconds** (2-byte unsigned integer, array size: ntach x nscan):
Tachometer sub-seconds. Values range from 0 to 62499 in units of 16 microseconds. The missing value is 65535.

**indexPulseSeconds** (4-byte unsigned integer, array size: nscan):
Index Pulse seconds. Values are in second. Special values are defined as:
0 Missing value

**indexPulseSubSeconds** (2-byte unsigned integer, array size: nscan):
Index Pulse subseconds. Values range from 0 to 62499 in units of 16 microseconds. The missing value is 65535.

**NEDTinfo** (Group in S1)

**NEDTinfo** (4-byte float, array size: nchan1 x nscan):
NEDT (Noise Equivalent Differential Temperature) for each channel.

**navigation** (Group in S1)

- **scPos** (4-byte float, array size: XYZ x nscan): The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
  -9999.9 Missing value

- **scVel** (4-byte float, array size: XYZ x nscan): The velocity vector \((ms^{-1})\) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
  -9999.9 Missing value

- **scLat** (4-byte float, array size: nscan): The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
  -9999.9 Missing value

- **scLon** (4-byte float, array size: nscan): The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
  -9999.9 Missing value

- **scAlt** (4-byte float, array size: nscan): The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
  -9999.9 Missing value

- **dprAlt** (4-byte float, array size: nscan): The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
  -9999.9 Missing value

- **scAttRollGeoc** (4-byte float, array size: nscan): The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity
direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:

-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:

-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:

-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range
from 0 to 10000000000 s. Special values are defined as:
-9999.9  Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9  Missing value

**nav2** (Group in S1)

**SCE_SELECTION** (2-byte unsigned integer, array size: nscan):
The current SCE selection setting. Special values are defined as:
0  Missing value

**SCE_RATE** (2-byte unsigned integer, array size: nscan):
The SMA rotational rate reported by the SCE. To obtain the spin rate in RPM, multiply **SCE_RATE** by 0.002999106 Values range from 1 to 65535 count. Special values are defined as:
0  Missing value

**calibration** (Group in S1)

**hotLoadTemp** (4-byte float, array size: nchan1 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray temperature and averaged over closest 5 scans Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**coldSkyTemp** (4-byte float, array size: nchan1 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**onOrbitNonLinearity** (4-byte float, array size: nchan1 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table . Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**derivedNonLinearity** (4-byte float, array size: nchan1 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special
values are defined as:
-9999.9 Missing value

**meanHotLoadCount** (4-byte float, array size: nchan1 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Hot Load Plus Noise Diode counts Averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCount** (4-byte float, array size: nchan1 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**diodeExcessTemp** (4-byte float, array size: nchan1 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature=diodExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature=diodExcessTemp + hotLoadTemp. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**gain** (4-byte float, array size: LNL x nchan1 x nscan):
gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9 Missing value

**offset** (4-byte float, array size: LNL x nchan1 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan1 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9 Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
calibrationQCflag. value 0 indicates good calibration. Values range from 0 to 15. Special values are defined as:
-9999 Missing value

**diodeFlag** (2-byte integer, array size: nscan):

Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

receiverTemp (4-byte float, array size: nchan1 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

receiverGain (4-byte float, array size: nchan1 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9 Missing value

cal2 (Group in S1)

trayTemperatureCount (2-byte unsigned integer, array size: nscan):
Counts to derive hot load tray temperature. Values range from 0 to 65535 count. Special values are defined as:
65535 Missing value

trayTemperature (4-byte float, array size: nscan):
Derive hot load tray temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

moonIndex (2-byte unsigned integer, array size: nchan1 x nscan):
Index determined by the angle between moon vector and cold sample vectors. 0 means angles between moon vector and all cold view vectors are greater than 5 degrees. Non-zero value means the number of cold samples that may be contaminated by moon. Values range from 0 to 100. Special values are defined as:
0 Missing value

noiseDiodeTemp (4-byte float, array size: nndiode x nscan):
Physical temperature of noise diode. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

TEMP_CALRES_2 (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for PRT temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0 Missing value

TEMP_CALRES_1 (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for PRT temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0 Missing value
RS_CALRES_2 (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
- 0 Missing value
RS_CALRES_1 (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
- 0 Missing value
BATC_CALRES_2 (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
- 0 Missing value
BATC_CALRES_1 (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
- 0 Missing value
NDIODE_MODE (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Mode. 0 = On every scan, 1 = On every other scan, 2 = Off. Values range from 0 to 2 count. Special values are defined as:
- 65535 Missing value
RSST_NDIODE_ST (2-byte unsigned integer, array size: nscan):
Noise diode state during the scan. 0 = Noise diodes OFF for the scan, 1 = Noise diodes ON for the scan. Values range from 0 to 1 count. Special values are defined as:
- 65535 Missing value
NDIODE10GHZNUM (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Start Sample Number, i.e., the sample number where noise diodes are turned on. Values range from 0 to 500 count. Special values are defined as:
- 65535 Missing value
hotLoadThermisterTemp (4-byte float, array size: ntherm x nchan1 x nscan):
Hot Load Thermister Temperature of 11 PRTs. Values range from 0 to 400 K. Special values are defined as:
- -9999.9 Missing value
TEMP_CALRES_4 (2-byte unsigned integer, array size: nscan):
Low calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
- 0 Missing value
TEMP_CALRES_5 (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
- 0 Missing value
TEMP_CALRES_6 (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
- 0  Missing value

TEMP_89GHZ_LO (4-byte float, array size: nscan):
89 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9  Missing value

TEMP_166GHZ_LO (4-byte float, array size: nscan):
166 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9  Missing value

TEMP_183GHZ_LO (4-byte float, array size: nscan):
183 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9  Missing value

TEMP_V89GHZMXR (4-byte float, array size: nscan):
89 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9  Missing value

TEMP_H89GHZMXR (4-byte float, array size: nscan):
89 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9  Missing value

TEMP_V166GHZMXR (4-byte float, array size: nscan):
166 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9  Missing value

TEMP_H166GHZMXR (4-byte float, array size: nscan):
166 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9  Missing value

TEMP_183GHZMXR (4-byte float, array size: nscan):
183 GHZ Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
- -9999.9  Missing value

TEMP_RS_MR1 (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 1 Values range from 0 to 500 K. Special values are defined as:
- -9999.9  Missing value
TEMP_RS_MR2 (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 2 Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

MR_ICA_TEMP (4-byte float, array size: nscan):
Main Reflector Temperature Read By ICA Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

MR_LR_LEFT_TEMP (4-byte float, array size: nscan):
Main Reflector left Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

MR_LR_RGHT_TEMP (4-byte float, array size: nscan):
Main Reflector right Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

MR_LR_LOWR_TEMP (4-byte float, array size: nscan):
Main Reflector lower Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

CSR_TEMP1 (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 1 Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

CSR_TEMP2 (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 2 Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

onOrbitDiodeExcessTemp (4-byte float, array size: nchan1 x nscan):
Diode Excess Temperature derived from on orbit trended look-up tables as a function of noise diode temperature from telemetry. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

WarmIntrusionToColdViewIndex (2-byte unsigned integer, array size: nchan1 x ncolds1 x nscan):
Index flag to determine if a cold view sample is contaminated by certain warmer sources. If the value is 0, the sample is good and the count is used in calibration. If the value is non-zero, the sample is contaminated and excluded in calibration.

0:  Good sample
1:  Bad sample determined by limit check
2: Bad sample determined by 2D medium filter

Values range from 0 to 2. Special values are defined as:
- 65535 Missing value

**moonVectorInstFrame** (4-byte float, array size: GMIxyz x nscan):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
- -9999.9 Missing value

**calCounts** (Group in S1)

**hotLoadReading** (2-byte unsigned integer, array size: nhots1 x nchan1 x nscan):
Hot Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
- 0 Missing value

**coldLoadReading** (2-byte unsigned integer, array size: ncolds1 x nchan1 x nscan):
Cold Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
- 0 Missing value

**hotLoadnDiodeReading** (2-byte unsigned integer, array size: nhots1 x nchan1 x nscan):
Hot Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
- 0 Missing value

**coldLoadnDiodeReading** (2-byte unsigned integer, array size: ncolds1 x nchan1 x nscan):
Cold Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
- 0 Missing value

**hotLoadThermisterCount** (2-byte unsigned integer, array size: ntherm x nscan):
Counts from 11 PRTs in the hot load. Values range from 0 to 65534 count. Special values are defined as:
- 65535 Missing value

**sunData** (Group in S1)

**solarBetaAngle** (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
- -9999.9 Missing value
**phaseFromOrbitMidnight** (4-byte float, array size: nscan):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of
the Earth center from the spacecraft and positive toward the spacecraft velocity direction
so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees
occurs with the spacecraft over the Earth's dawn terminator, 180 degrees occurs at local
orbit noon, and -90 degrees occurs with the spacecraft over the Earth's dusk terminator.
Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

**sunEarthSeparation** (4-byte float, array size: nscan):
The separation angle between the Sun and Earth directions from the spacecraft. Values
range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

**earthAngularRadius** (4-byte float, array size: nscan):
The angle between the center of the Earth and the horizon edge. The sun is above
the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius.
Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**phaseOfEclipseExit** (4-byte float, array size: nscan):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow,
based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from
0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**orbitRate** (4-byte float, array size: nscan):
The instantaneous angular rate of the spacecraft around the orbit. Values range from
0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth's shadow. Values
range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such
that +Z is nominally toward the Earth and gives the instrument spin axis, and data is
collected nominally centered about the +X direction. Values range from 0 to 1.0. Special
values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npix1 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel
location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value
satAzimuthAngle (4-byte float, array size: npix1 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarZenAngle (4-byte float, array size: npix1 x nscan):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarAzimuthAngle (4-byte float, array size: npix1 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (4-byte float, array size: npix1 x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

magneticFieldVector (4-byte float, array size: GMIxyz x nscan):
Magnetometer volt reading in TAM (x, y, z) coordinate system. Used to perform along-scan correction of earth view counts. (The TAM (x,y,z) coordinate system is similar to GPM S/C coordinate system but y and z axis are rotated by 180 degrees.) Values range from -500 to 500 V. Special values are defined as:
-9999.9 Missing value

TAMmagneticFieldVector (4-byte float, array size: GMIxyz x nscan):
Magnetic Field derived from GPM three-axis magnetometer (TAM). Values range from -1000 to 1000 V. Special values are defined as:
-9999.9 Missing value

earthViewCounts (2-byte unsigned integer, array size: nchan1 x npix1 x nscan):
Earth view counts. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

Tb (4-byte float, array size: nchan1 x npix1 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

RFIFlag (2-byte integer, array size: nfreq1 x npix1 x nscan):
Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: Not affected by RFI.
1: Affected by RFI with X-cal filter.
2: Affected by RFI with RSS filter.
3-7: Spare
-9999: Missing

S2 (Swath)

S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value
MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npix2 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npix2 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S2)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits = 0, the unsigned integer value is $2^i$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
</tbody>
</table>
5.6 1BASEGMRSS - GMI Antenna Temperatures

4 Housekeeping (HK) telemetry packet missing
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
</tbody>
</table>
STANDARD GPM PRODUCTS

8   Error in getting spacecraft attitude for any pixel
9   Error in getting spacecraft ephemeris for any pixel
10  Spare (always 0)
11  Spare (always 0)
12  Spare (always 0)
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector \(v\) from the satellite forward direction of motion, measured clockwise facing down. We define \(v\) in the same direction as the spacecraft axis \(+X\), which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
</tbody>
</table>
pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
Status of the GMI instrument.
Bit Meaning if bit = 1
0 Receiver status (0=ON, 1=OFF)
1 Spinup Status (0=ON, 1=OFF)

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**sampleHeader** (Group in S2)

**blanking** (1-byte integer, array size: nscan):
Value of 0 = Blanking off
Value of 1 = Blanking on

**earthViewFirstSample** (2-byte integer, array size: nscan):
Sample number of the first earth view. Values range from 0 to 512. Special values are
declared as:
-9999 Missing value

**sampleNumber** (2-byte integer, array size: nsamt x nchan2 x nscan):
Number of valid samples in scan. Values range from 0 to 512. Special values are defined
as:
-9999 Missing value

**tachSeconds** (4-byte unsigned integer, array size: ntach x nscan):
Tachometer seconds. Values are in second. Special values are defined as:
0 Missing value

**tachSubSeconds** (2-byte unsigned integer, array size: ntach x nscan):
Tachometer subseconds. Values range from 0 to 62499 in units of 16 microseconds. The
missing value is 65535.

**indexPulseSeconds** (4-byte unsigned integer, array size: nscan):
Index Pulse seconds. Values are in second. Special values are defined as:
0 Missing value

**indexPulseSubSeconds** (2-byte unsigned integer, array size: nscan):
Index Pulse subseconds. Values range from 0 to 62499 in units of 16 microseconds. The
missing value is 65535.
NEDTinfo (Group in S2)

NEDTinfo (4-byte float, array size: nchan2 x nscan):

NEDT (Noise Equivalent Differential Temperature) for each channel.

navigation (Group in S2)

scPos (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector ($ms^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed
using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:

-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:

-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coor-
coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980.
timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

nav2 (Group in S2)

SCE_SELECTION (2-byte unsigned integer, array size: nscan):
The current SCE selection setting. Special values are defined as:
0 Missing value

SCE_RATE (2-byte unsigned integer, array size: nscan):
The SMA rotational rate reported by the SCE. To obtain the spin rate in RPM, multiply SCE_RATE by 0.002999106 Values range from 1 to 65535 count. Special values are defined as:
0 Missing value

calibration (Group in S2)

hotLoadTemp (4-byte float, array size: nchan2 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray temperature and averaged over closest 5 scans Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

coldSkyTemp (4-byte float, array size: nchan2 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

onOrbitNonLinearity (4-byte float, array size: nchan2 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table. Values
range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**derivedNonLinearity** (4-byte float, array size: nchan2 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**meanHotLoadCount** (4-byte float, array size: nchan2 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Hot Load Plus Noise Diode counts. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**meanColdSkyCount** (4-byte float, array size: nchan2 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**diodeExcessTemp** (4-byte float, array size: nchan2 x nscan):
The Noise Diode Excess Temperature. Cold and diode coupled Temperature=diodeExcessTemp + coldSkyTemp. Hot and diode coupled Temperature=diodeExcessTemp + hotLoadTemp. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**gain** (4-byte float, array size: LNL x nchan2 x nscan):
gain[0] determine the total Ta gain. 
Ta=offset[0]+gain[0]*earthCount+nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9  Missing value

**offset** (4-byte float, array size: LNL x nchan2 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan2 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9  Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
calibrationQCflag. value 0 indicates good calibration. Values range from 0 to 15. Special
values are defined as:
-9999  Missing value

**diodeFlag** (2-byte integer, array size: nscan):

Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

**receiverTemp** (4-byte float, array size: nchan2 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**receiverGain** (4-byte float, array size: nchan2 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9  Missing value

**cal2** (Group in S2)

**trayTemperatureCount** (2-byte unsigned integer, array size: nscan):
Counts to derive hot load tray temperature. Values range from 0 to 65534 count. Special values are defined as:
65535  Missing value

**trayTemperature** (4-byte float, array size: nscan):
Derive hot load tray temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**moonIndex** (2-byte unsigned integer, array size: nchan2 x nscan):
Index determined by the angle between moon vector and cold sample vectors. 0 means angles between moon vector and all cold view vectors are greater than 5 degrees. Non-zero value means the number of cold samples that may be contaminated by moon. Values range from 0 to 100. Special values are defined as:
0  Missing value

**noiseDiodeTemp** (4-byte float, array size: nndiode x nscan):
Physical temperature of noise diode. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**TEMP_CALRES_2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for PRT temperature retrieval. Values range from
0 to 65535 count. Special values are defined as:
0    Missing value

**TEMP_CALRES_1** (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for PRT temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0    Missing value

**RS_CALRES_2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0    Missing value

**RS_CALRES_1** (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0    Missing value

**BATC_CALRES_2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0    Missing value

**BATC_CALRES_1** (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0    Missing value

**NDIODE_MODE** (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Mode. 0 = On every scan, 1 = On every other scan, 2 = Off. Values range from 0 to 2 count. Special values are defined as:
65535    Missing value

**RSST_NDIODE_ST** (2-byte unsigned integer, array size: nscan):
Noise diode state during the scan. 0 = Noise diodes OFF for the scan, 1 = Noise diodes ON for the scan. Values range from 0 to 1 count. Special values are defined as:
65535    Missing value

**NDIODE10GHZNUM** (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Start Sample Number, i.e., the sample number where noise diodes are turned on. Values range from 0 to 500 count. Special values are defined as:
65535    Missing value

**hotLoadThermisterTemp** (4-byte float, array size: ntherm x nchan2 x nscan):
Hot Load Thermister Temperature of 11 PRTs. Values range from 0 to 400 K. Special values are defined as:
-9999.9    Missing value
TEMP_CALRES_4 (2-byte unsigned integer, array size: nscan):
Low calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
0  Missing value

TEMP_CALRES_5 (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
0  Missing value

TEMP_CALRES_6 (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
0  Missing value

TEMP_89GHZ_LO (4-byte float, array size: nscan):
89 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP_166GHZ_LO (4-byte float, array size: nscan):
166 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP_183GHZ_LO (4-byte float, array size: nscan):
183 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP_V89GHZMXR (4-byte float, array size: nscan):
89 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP_H89GHZMXR (4-byte float, array size: nscan):
89 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP_V166GHZMXR (4-byte float, array size: nscan):
166 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP_H166GHZMXR (4-byte float, array size: nscan):
166 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value
**TEMP\_183GHZMXR** (4-byte float, array size: nscan):
183 GHZ Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

**TEMP\_RS\_MR1** (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 1 Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

**TEMP\_RS\_MR2** (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 2 Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

**MR\_ICA\_TEMP** (4-byte float, array size: nscan):
Main Reflector Temperature Read By ICA Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

**MR\_LR\_LEFT\_TEMP** (4-byte float, array size: nscan):
Main Reflector left Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

**MR\_LR\_RGHT\_TEMP** (4-byte float, array size: nscan):
Main Reflector right Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

**MR\_LR\_LOWR\_TEMP** (4-byte float, array size: nscan):
Main Reflector lower Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

**CSR\_TEMP1** (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 1 Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

**CSR\_TEMP2** (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 2 Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

**onOrbitDiodeExcessTemp** (4-byte float, array size: nchan2 x nscan):
Diode Excess Temperature derived from on orbit trended look-up tables as a function of noise diode temperature from telemetry. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value
**WarmIntrusionToColdViewIndex** (2-byte unsigned integer, array size: nchan2 x ncolds2 x nscan):
Index flag to determine if a cold view sample is contaminated by certain warmer sources. If the value is 0, the sample is good and the count is used in calibration. If the value is non-zero, the sample is contaminated and excluded in calibration.

0: Good sample  
1: Bad sample determined by limit check  
2: Bad sample determined by 2D medium filter 

Values range from 0 to 2. Special values are defined as:
65535 Missing value

**moonVectorInstFrame** (4-byte float, array size: GMIxyz x nscan):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
-9999.9 Missing value

**calCounts** (Group in S2)

**hotLoadReading** (2-byte unsigned integer, array size: nhots2 x nchan2 x nscan):
Hot Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**coldLoadReading** (2-byte unsigned integer, array size: ncolds2 x nchan2 x nscan):
Cold Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**hotLoadnDiodeReading** (2-byte unsigned integer, array size: nhots2 x nchan2 x nscan):
Hot Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**coldLoadnDiodeReading** (2-byte unsigned integer, array size: ncolds2 x nchan2 x nscan):
Cold Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

**hotLoadThermisterCount** (2-byte unsigned integer, array size: ntherm x nscan):
Counts from 11 PRTs in the hot load Values range from 0 to 65534 count. Special values are defined as:
65535 Missing value
sunData (Group in S2)

solarBetaAngle (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nscan):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special
5.6  IBASEGMIRSS - GMI Antenna Temperatures

values are defined as:
-9999.9  Missing value

**incidenceAngle** (4-byte float, array size: npix2 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**satAzimuthAngle** (4-byte float, array size: npix2 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**solarZenAngle** (4-byte float, array size: npix2 x nscan):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**solarAzimuthAngle** (4-byte float, array size: npix2 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**sunGlintAngle** (4-byte float, array size: npix2 x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**magneticFieldVector** (4-byte float, array size: GMIxyz x nscan):
Magnetometer volt reading in TAM (x, y, z) coordinate system. Used to perform along-scan correction of earth view counts. (The TAM (x,y,z) coordinate system is similar to GPM S/C coordinate system but y and z axis are rotated by 180 degrees.) Values range from -500 to 500 V. Special values are defined as:
-9999.9  Missing value

**TAMmagneticFieldVector** (4-byte float, array size: GMIxyz x nscan):
Magnetic Field derived from GPM three-axis magnetometer (TAM). Values range from -1000 to 1000 V. Special values are defined as:
-9999.9  Missing value

**earthViewCounts** (2-byte unsigned integer, array size: nchan2 x npix2 x nscan):
Earth view counts. Values range from 0 to 65535 counts. Special values are defined as:
0  Missing value
\textbf{Tb} (4-byte float, array size: nchan2 x npix2 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
\begin{itemize}
  \item -9999.9 Missing value
\end{itemize}

\textbf{RFIFlag} (2-byte integer, array size: nfreq2 x npix2 x nscan):

Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:
\begin{itemize}
  \item 0: Not affected by RFI.
  \item 1: Affected by RFI with X-cal filter.
  \item 2: Affected by RFI with RSS filter.
  \item 3-7: Spare
  \item -9999: Missing
\end{itemize}

\textbf{S3} (Swath)

\textbf{S3_SwathHeader} (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

\textbf{ScanTime} (Group in S3)
A UTC time associated with the scan.

\textbf{Year} (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
\begin{itemize}
  \item -9999 Missing value
\end{itemize}

\textbf{Month} (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
\begin{itemize}
  \item -99 Missing value
\end{itemize}

\textbf{DayOfMonth} (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
\begin{itemize}
  \item -99 Missing value
\end{itemize}

\textbf{Hour} (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
\begin{itemize}
  \item -99 Missing value
\end{itemize}
5.6 1BASEGMIRSS - GMI Antenna Temperatures

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousands of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: npix3 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: npix3 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scanStatus (Group in S3)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>
missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
</tbody>
</table>
5.6 1BASEGMIRSS - GMI Antenna Temperatures

1. Negative scan time, invalid input
2. Error getting spacecraft attitude at scan mid-time
3. Error getting spacecraft ephemeris at scan mid-time
4. Invalid input non-unit ray vector for any pixel
5. Ray misses Earth for any pixel with normal pointing
6. Nadir calculation error for subsatellite position
7. Pixel count with geolocation error over threshold
8. Error in getting spacecraft attitude for any pixel
9. Error in getting spacecraft ephemeris for any pixel
10. Spare (always 0)
11. Spare (always 0)
12. Spare (always 0)
13. Spare (always 0)
14. Spare (always 0)
15. Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of mo-
tion, measured clockwise facing down. We define $v$ in the same direction as the spacecraft axis $+X$, which is also the center of the GMI scan. If $SCorientation$ is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
</tbody>
</table>
2 S/C Z axis nadir, -X in flight direction
3 Flight Z axis nadir, -X in flight direction
4 +90 yaw for DPR antenna pattern calibration
5 -90 yaw for DPR antenna pattern calibration
-99 Missing

**operationalMode** (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**calibration** (Group in S3)

**hotLoadTemp** (4-byte float, array size: nchan1 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray temperature and averaged over closest 5 scans. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**coldSkyTemp** (4-byte float, array size: nchan1 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**onOrbitNonLinearity** (4-byte float, array size: nchan1 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**derivedNonLinearity** (4-byte float, array size: nchan1 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value
**meanHotLoadCount** (4-byte float, array size: nchan1 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Hot Load Plus Noise Diode counts averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCount** (4-byte float, array size: nchan1 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**diodeExcessTemp** (4-byte float, array size: nchan1 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature = diodeExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature = diodeExcessTemp + hotLoadTemp. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**gain** (4-byte float, array size: LNL x nchan1 x nscan):
Gain[0] determine the total T gain. Tgain = offset[0] + gain[0]*earthCount + nonLinearGain*earthCount*earthCount. Nonlinearity = offset[1] + gain[1]*earthCount + nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9 Missing value

**offset** (4-byte float, array size: LNL x nchan1 x nscan):
Offset[0] determine the total T offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan1 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9 Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
Calibration QCflag. value 0 indicates good calibration. Values range from 0 to 15. Special values are defined as:
-9999 Missing value

**diodeFlag** (2-byte integer, array size: nscan):
Diode flag.
0, Noise Diode on
2. Noise Diode off
5. Noise Diode status unknown

**receiverTemp** (4-byte float, array size: nchan1 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**receiverGain** (4-byte float, array size: nchan1 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npix3 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Tb** (4-byte float, array size: nchan1 x npix3 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**count** (2-byte unsigned integer, array size: nchan1 x npix3 x nscan):
Full scan count. Values range from 0 to 65534. Special values are defined as:
65535 Missing value

**S4** (Swath)

**S4_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S4)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value
DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npix x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npix x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S4)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).
Bit Meaning if bit = 1
0  missing
5  geoError is not zero
6  modeStatus is not zero

missing  (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

Bit Meaning if bit = 1
0  Scan is missing
1  Science telemetry packet missing
2  Science telemetry segment within packet missing
3  Science telemetry other missing
4  Housekeeping (HK) telemetry packet missing
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

modeStatus  (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit =
1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other
bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

Bit Meaning if bit = 1
0  Spare (always 0)
1  SCorientation not 0 or 180
2  pointingStatus not 0
3  Spare (always 0)
4  Non-routine operationalMode
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

geoError  (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
14  Spare (always 0)
15  Spare (always 0)

**SCorientation** (2-byte integer, array size: nscan): The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan): pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**acsModeMidScan** (1-byte integer, array size: nscan): acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: nscan): targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.
<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):
Status of the GMI instrument.

- Bit Meaning if bit = 1
  - 0 Receiver status (0=ON, 1=OFF)
  - 1 Spinup Status (0=ON, 1=OFF)

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
- 9999.9 Missing value

**calibration** (Group in S4)

**hotLoadTemp** (4-byte float, array size: nchan2 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray temperature and averaged over closest 5 scans. Values range from 0 to 400 K. Special values are defined as:
- 9999.9 Missing value

**coldSkyTemp** (4-byte float, array size: nchan2 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
- 9999.9 Missing value

**onOrbitNonLinearity** (4-byte float, array size: nchan2 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table. Values range from 0 to 400 K. Special values are defined as:
- 9999.9 Missing value
**derivedNonLinearity** (4-byte float, array size: nchan2 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCount** (4-byte float, array size: nchan2 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Hot Load Plus Noise Diode counts Averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCount** (4-byte float, array size: nchan2 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**diodeExcessTemp** (4-byte float, array size: nchan2 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature=diodeExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature=diodeExcessTemp + hotLoad-Temp. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**gain** (4-byte float, array size: LNL x nchan2 x nscan):
gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9 Missing value

**offset** (4-byte float, array size: LNL x nchan2 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan2 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9 Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
calibrationQCflag. value 0 indicates good calibration. Values range from 0 to 15. Special values are defined as:
-9999 Missing value
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
  -9999.9 Missing value

 receiverGain (4-byte float, array size: nchan2 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
  -9999.9 Missing value

 incidenceAngle (4-byte float, array size: npix4 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel
location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
  -9999.9 Missing value

 Tb (4-byte float, array size: nchan2 x npix4 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
  -9999.9 Missing value

 count (2-byte unsigned integer, array size: nchan2 x npix4 x nscan):
Full scan count. Values range from 0 to 65534. Special values are defined as:
  65535 Missing value

**C Structure Header file:**

```c
#ifndef _L1BASEGMRSS_S4_CALIBRATION_
#define _L1BASEGMRSS_S4_CALIBRATION_

typedef struct {
    float hotLoadTemp[4];
    float coldSkyTemp[4];
    float onOrbitNonLinearity[4];
    float derivedNonLinearity[4];
    float meanHotLoadCount[4];
    float meanHotLoadCntnDiode[4];
    float meanColdSkyCount[4];
    float meanColdSkyCntnDiode[4];
} _L1BASEGMRSS_S4_CALIBRATION_
```
typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMIRSS_S4_SCANSTATUS;

#endif

#ifndef _L1BASEGMIRSS_S4_
#define _L1BASEGMIRSS_S4_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[500];
    float Longitude[500];
    L1BASEGMIRSS_S4_SCANSTATUS scanStatus;
    L1BASEGMIRSS_S4_CALIBRATION calibration;
    float incidenceAngle[500];
    float Tb[500][4];
    unsigned short count[500][4];
} L1BASEGMIRSS_S4;
typedef struct {
    float hotLoadTemp[9];
    float coldSkyTemp[9];
    float onOrbitNonLinearity[9];
    float derivedNonLinearity[9];
    float meanHotLoadCount[9];
    float meanHotLoadCntnDiode[9];
    float meanColdSkyCount[9];
    float meanColdSkyCntnDiode[9];
    float diodeExcessTemp[9];
    float gain[9][2];
    float offset[9][2];
    float nonLinearGain[9];
    short calibrationQCflag;
    short diodeFlag;
    float receiverTemp[9];
    float receiverGain[9];
} L1BASEGMIRSS_S3_CALIBRATION;

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMIRSS_S3_SCANSTATUS;
typedef struct {
    SCANTIME ScanTime;
    float Latitude[500];
    float Longitude[500];
    L1BASEGMIRSS_S3_SCANSTATUS scanStatus;
    L1BASEGMIRSS_S3_CALIBRATION calibration;
    float incidenceAngle[500];
    float Tb[500][9];
    unsigned short count[500][9];
} L1BASEGMIRSS_S3;

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1BASEGMIRSS_S2_SUNDATA;

typedef struct {
    unsigned short hotLoadReading[4][65];
    unsigned short coldLoadReading[4][85];
    unsigned short hotLoadnDiodeReading[4][65];
} L1BASEGMIRSS_S2_CALCOUNTS;
unsigned short coldLoadnDiodeReading[4][85];
unsigned short hotLoadThermisterCount[11];
} L1BASEGMIRSS_S2_CALCOUNTS;

#endif

#ifndef _L1BASEGMIRSS_S2_CAL2_
define _L1BASEGMIRSS_S2_CAL2_

typedef struct {
  unsigned short trayTemperatureCount;
  float trayTemperature;
  unsigned short moonIndex[4];
  float noiseDiodeTemp[6];
  unsigned short TEMP_CALRES_2;
  unsigned short TEMP_CALRES_1;
  unsigned short RS_CALRES_2;
  unsigned short RS_CALRES_1;
  unsigned short BATC_CALRES_2;
  unsigned short BATC_CALRES_1;
  unsigned short NDIODE_MODE;
  unsigned short RSST_NDIODE_ST;
  unsigned short NDIODE10GHZNUM;
  float hotLoadThermisterTemp[4][11];
  unsigned short TEMP_CALRES_4;
  unsigned short TEMP_CALRES_5;
  unsigned short TEMP_CALRES_6;
  float TEMP_89GHZ_LO;
  float TEMP_166GHZ_LO;
  float TEMP_183GHZ_LO;
  float TEMP_V89GHZMXR;
  float TEMP_H89GHZMXR;
  float TEMP_V166GHZMXR;
  float TEMP_H166GHZMXR;
  float TEMP_183GHZMXR;
  float TEMP_RS_MR1;
  float TEMP_RS_MR2;
  float MR_ICA_TEMP;
  float MR_LR_LEFT_TEMP;
  float MR_LR_RGHT_TEMP;
  float MR_LR_LOWR_TEMP;
  float CSR_TEMP1;
  float CSR_TEMP2;
}
float onOrbitDiodeExcessTemp[4];
unsigned short WarmIntrusionToColdViewIndex[85][4];
} L1BASEGMIRSS_S2_CAL2;

#endif

#ifndef _L1BASEGMIRSS_S2_CALIBRATION_
#define _L1BASEGMIRSS_S2_CALIBRATION_

typedef struct {
  float hotLoadTemp[4];
  float coldSkyTemp[4];
  float onOrbitNonLinearity[4];
  float derivedNonLinearity[4];
  float meanHotLoadCount[4];
  float meanHotLoadCntnDiode[4];
  float meanColdSkyCount[4];
  float meanColdSkyCntnDiode[4];
  float diodeExcessTemp[4];
  float gain[4][2];
  float offset[4][2];
  float nonLinearGain[4];
  short calibrationQCflag;
  short diodeFlag;
  float receiverTemp[4];
  float receiverGain[4];
} L1BASEGMIRSS_S2_CALIBRATION;
#endif

#ifndef _L1BASEGMIRSS_S2_NAV2_
#define _L1BASEGMIRSS_S2_NAV2_

typedef struct {
  unsigned short SCE_SELECTION;
  unsigned short SCE_RATE;
} L1BASEGMIRSS_S2_NAV2;
#endif

#ifndef _L1BASEGMIRSS_S2_NEDTINFO_
#define _L1BASEGMIRSS_S2_NEDTINFO_

typedef struct {
    float NEDTinfo[4];
} L1BASEGMIRSS_S2_NEDTINFO;
#endif

#ifndef _L1BASEGMIRSS_S2_SAMPLEHEADER_
define _L1BASEGMIRSS_S2_SAMPLEHEADER_

typedef struct {
    signed char blanking;
    short earthViewFirstSample;
    short sampleNumber[4][4];
    unsigned int tachSeconds[32];
    unsigned short tachSubSeconds[32];
    unsigned int indexPulseSeconds;
    unsigned short indexPulseSubSeconds;
} L1BASEGMIRSS_S2_SAMPLEHEADER;
#endif

#ifndef _L1BASEGMIRSS_S2_SCANSTATUS_
define _L1BASEGMIRSS_S2_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMIRSS_S2_SCANSTATUS;
#endif

#ifndef _L1BASEGMIRSS_S2_
define _L1BASEGMIRSS_S2_


typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    L1BASEGMIRSS_S2_SCANSTATUS scanStatus;
    L1BASEGMIRSS_S2_SAMPLEHEADER sampleHeader;
    L1BASEGMIRSS_S2_NEDTINFO NEDTinfo;
    NAVIGATION navigation;
    L1BASEGMIRSS_S2_NAV2 nav2;
    L1BASEGMIRSS_S2_CALIBRATION calibration;
    L1BASEGMIRSS_S2_CAL2 cal2;
    float moonVectorInstFrame[3];
    L1BASEGMIRSS_S2_CALCOUNTS calCounts;
    L1BASEGMIRSS_S2_SUNDATA sunData;
    float incidenceAngle[221];
    float satAzimuthAngle[221];
    float solarZenAngle[221];
    float solarAzimuthAngle[221];
    float sunGlintAngle[221];
    float magneticFieldVector[3];
    float TAMmagneticFieldVector[3];
    unsigned short earthViewCounts[221][4];
    float Tb[221][4];
    short RFIFlag[221][2];
} L1BASEGMIRSS_S2;

#endif

#ifndef _L1BASEGMIRSS_S1_SUNDATA_
#define _L1BASEGMIRSS_S1_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1BASEGMIRSS_S1_SUNDATA;

#endif
#ifndef _L1BASEGMRSS_S1_CALCOUNTS_
define _L1BASEGMRSS_S1_CALCOUNTS_

typedef struct {
    unsigned short hotLoadReading[9][65];
    unsigned short coldLoadReading[9][85];
    unsigned short hotLoadnDiodeReading[9][65];
    unsigned short coldLoadnDiodeReading[9][85];
    unsigned short hotLoadThermisterCount[11];
} L1BASEGMRSS_S1_CALCOUNTS;
#endif

#else
#endif

#define _L1BASEGMRSS_S1_CAL2_

define _L1BASEGMRSS_S1_CAL2_

typedef struct {
    unsigned short trayTemperatureCount;
    float trayTemperature;
    unsigned short moonIndex[9];
    float noiseDiodeTemp[6];
    unsigned short TEMP_CALRES_2;
    unsigned short TEMP_CALRES_1;
    unsigned short RS_CALRES_2;
    unsigned short RS_CALRES_1;
    unsigned short BATC_CALRES_2;
    unsigned short BATC_CALRES_1;
    unsigned short NDIODE_MODE;
    unsigned short RSST_NDIODE_ST;
    unsigned short NDIODE10GHZNUM;
    float hotLoadThermisterTemp[9][11];
    unsigned short TEMP_CALRES_4;
    unsigned short TEMP_CALRES_5;
    unsigned short TEMP_CALRES_6;
    float TEMP_89GHZ_LO;
    float TEMP_166GHZ_LO;
    float TEMP_183GHZ_LO;
    float TEMP_V89GHZMXR;
    float TEMP_H89GHZMXR;
    float TEMP_V166GHZMXR;
    float TEMP_H166GHZMXR;
    float TEMP_183GHZMXR;
} L1BASEGMRSS_S1_CAL2;
#endif
5.6 1BASEGMIRSS - GMI Antenna Temperatures

float TEMP_RS_MR1;
float TEMP_RS_MR2;
float MR_ICA_TEMP;
float MR_LR_LEFT_TEMP;
float MR_LR_RIGHT_TEMP;
float MR_LR_LOW_R_TEMP;
float CSR_TEMP1;
float CSR_TEMP2;
float onOrbitDiodeExcessTemp[9];
unsigned short WarmIntrusionToColdViewIndex[85][9];

} L1BASEGMIRSS_S1_CAL2;

#endif

#ifndef _L1BASEGMIRSS_S1_CALIBRATION_
#define _L1BASEGMIRSS_S1_CALIBRATION_

typedef struct {
    float hotLoadTemp[9];
    float coldSkyTemp[9];
    float onOrbitNonLinearity[9];
    float derivedNonLinearity[9];
    float meanHotLoadCount[9];
    float meanHotLoadCntnDiode[9];
    float meanColdSkyCount[9];
    float meanColdSkyCntnDiode[9];
    float diodeExcessTemp[9];
    float gain[9][2];
    float offset[9][2];
    float nonLinearGain[9];
    short calibrationQCflag;
    short diodeFlag;
    float receiverTemp[9];
    float receiverGain[9];
} L1BASEGMIRSS_S1_CALIBRATION;

#endif

#ifndef _L1BASEGMIRSS_S1_NAV2_
#define _L1BASEGMIRSS_S1_NAV2_

typedef struct {
    unsigned short SCE_SELECTION;


unsigned short SCE_RATE;
} L1BASEGMRSS_S1_NAV2;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;
#endif

#ifndef _L1BASEGMRSS_S1_NEDTINFO_
#define _L1BASEGMRSS_S1_NEDTINFO_

typedef struct {
    float NEDTinfo[9];
} L1BASEGMRSS_S1_NEDTINFO;
#endif

#ifndef _L1BASEGMRSS_S1_SAMPLEHEADER_
#define _L1BASEGMRSS_S1_SAMPLEHEADER_

typedef struct {
    signed char blanking;
    short earthViewFirstSample;
} L1BASEGMRSS_S1_SAMPLEHEADER;
#endif
short sampleNumber[9][4];
unsigned int tachSeconds[32];
unsigned short tachSubSeconds[32];
unsigned int indexPulseSeconds;
unsigned short indexPulseSubSeconds;
} L1BASEGMIRSS_S1_SAMPLEHEADER;

#endif

#ifndef _L1BASEGMIRSS_S1_SCANSTATUS_
#define _L1BASEGMIRSS_S1_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMIRSS_S1_SCANSTATUS;

#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#ifndef _L1BASEGMIRSS_S1_
#define _L1BASEGMIRSS_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    L1BASEGMIRSS_S1_SCANSTATUS scanStatus;
    L1BASEGMIRSS_S1_SAMPLEHEADER sampleHeader;
    L1BASEGMIRSS_S1_NEDTINFO NEDTinfo;
    NAVIGATION navigation;
    L1BASEGMIRSS_S1_NAV2 nav2;
    L1BASEGMIRSS_S1_CALIBRATION calibration;
    L1BASEGMIRSS_S1_CAL2 cal2;
    float moonVectorInstFrame[3];
    L1BASEGMIRSS_S1_CALCOUNTS calCounts;
    L1BASEGMIRSS_S1_SUNDATA sunData;
    float incidenceAngle[221];
    float satAzimuthAngle[221];
    float solarZenAngle[221];
    float solarAzimuthAngle[221];
    float sunGlintAngle[221];
    float magneticFieldVector[3];
    float TAMmagneticFieldVector[3];
    unsigned short earthViewCounts[221][9];
    float Tb[221][9];
    short RFIFlag[221][5];
} L1BASEGMIRSS_S1;

#endif

#ifndef _L1BASEGMIRSS_SWATHS_
#define _L1BASEGMIRSS_SWATHS_

typedef struct {
    L1BASEGMIRSS_S1 S1;
    L1BASEGMIRSS_S2 S2;
    L1BASEGMIRSS_S3 S3;
    L1BASEGMIRSS_S4 S4;
} L1BASEGMIRSS_SWATHS;
Fortran Structure Header file:

```
STRUCTURE /L1BASEGMIRSS_S4_CALIBRATION/
  REAL*4 hotLoadTemp(4)
  REAL*4 coldSkyTemp(4)
  REAL*4 onOrbitNonLinearity(4)
  REAL*4 derivedNonLinearity(4)
  REAL*4 meanHotLoadCount(4)
  REAL*4 meanHotLoadCntnDiode(4)
  REAL*4 meanColdSkyCount(4)
  REAL*4 meanColdSkyCntnDiode(4)
  REAL*4 diodeExcessTemp(4)
  REAL*4 gain(2,4)
  REAL*4 offset(2,4)
  REAL*4 nonLinearGain(4)
  INTEGER*2 calibrationQCflag
  INTEGER*2 diodeFlag
  REAL*4 receiverTemp(4)
  REAL*4 receiverGain(4)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S4_SCANSTATUS/
  BYTE dataQuality
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S4/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(500)
  REAL*4 Longitude(500)
```
RECORD /L1BASEGMIRSS_S4_SCANSTATUS/ scanStatus
RECORD /L1BASEGMIRSS_S4_CALIBRATION/ calibration
REAL*4 incidenceAngle(500)
REAL*4 Tb(4,500)
INTEGER*2 count(4,500)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S3_CALIBRATION/
  REAL*4 hotLoadTemp(9)
  REAL*4 coldSkyTemp(9)
  REAL*4 onOrbitNonLinearity(9)
  REAL*4 derivedNonLinearity(9)
  REAL*4 meanHotLoadCount(9)
  REAL*4 meanHotLoadCntnDiode(9)
  REAL*4 meanColdSkyCount(9)
  REAL*4 meanColdSkyCntnDiode(9)
  REAL*4 diodeExcessTemp(9)
  REAL*4 gain(2,9)
  REAL*4 offset(2,9)
  REAL*4 nonLinearGain(9)
  INTEGER*2 calibrationQCflag
  INTEGER*2 diodeFlag
  REAL*4 receiverTemp(9)
  REAL*4 receiverGain(9)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S3_SCANSTATUS/
  BYTE dataQuality
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S3/
  RECORD /SCAN TIME/ ScanTime
  REAL*4 Latitude(500)
REAL*4 Longitude(500)
RECORD /L1BASEGMIRSS_S3_SCANSTATUS/ scanStatus
RECORD /L1BASEGMIRSS_S3_CALIBRATION/ calibration
REAL*4 incidenceAngle(500)
REAL*4 Tb(9,500)
INTEGER*2 count(9,500)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S2_SUNDATA/
   REAL*4 solarBetaAngle
   REAL*4 phaseFromOrbitMidnight
   REAL*4 sunEarthSeparation
   REAL*4 earthAngularRadius
   REAL*4 phaseOfEclipseExit
   REAL*4 orbitRate
   REAL*4 timeSinceEclipseEntry
   REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S2_CALCOUNTS/
   INTEGER*2 hotLoadReading(65,4)
   INTEGER*2 coldLoadReading(85,4)
   INTEGER*2 hotLoadnDiodeReading(65,4)
   INTEGER*2 coldLoadnDiodeReading(85,4)
   INTEGER*2 hotLoadThermisterCount(11)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S2_CAL2/
   INTEGER*2 trayTemperatureCount
   REAL*4 trayTemperature
   INTEGER*2 moonIndex(4)
   REAL*4 noiseDiodeTemp(6)
   INTEGER*2 TEMP_CALRES_2
   INTEGER*2 TEMP_CALRES_1
   INTEGER*2 RS_CALRES_2
   INTEGER*2 RS_CALRES_1
   INTEGER*2 BATC_CALRES_2
   INTEGER*2 BATC_CALRES_1
   INTEGER*2 NDIODE_MODE
   INTEGER*2 RSST_NDIODE_ST
   INTEGER*2 NDIODE10GHZNUM
   REAL*4 hotLoadThermisterTemp(11,4)
   INTEGER*2 TEMP_CALRES_4
INTEGER*2 TEMP_CALRES_5
INTEGER*2 TEMP_CALRES_6
REAL*4 TEMP_89GHZ_LO
REAL*4 TEMP_166GHZ_LO
REAL*4 TEMP_183GHZ_LO
REAL*4 TEMP_V89GHZMXR
REAL*4 TEMP_H89GHZMXR
REAL*4 TEMP_V166GHZMXR
REAL*4 TEMP_H166GHZMXR
REAL*4 TEMP_183GHZMXR
REAL*4 TEMP_RS_MR1
REAL*4 TEMP_RS_MR2
REAL*4 MR_ICA_TEMP
REAL*4 MR_LR_LEFT_TEMP
REAL*4 MR_LR_RGHT_TEMP
REAL*4 MR_LR_LOWR_TEMP
REAL*4 CSR_TEMP1
REAL*4 CSR_TEMP2
REAL*4 onOrbitDiodeExcessTemp(4)
INTEGER*2 WarmIntrusionToColdViewIndex(4,85)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S2_CALIBRATION/
REAL*4 hotLoadTemp(4)
REAL*4 coldSkyTemp(4)
REAL*4 onOrbitNonLinearity(4)
REAL*4 derivedNonLinearity(4)
REAL*4 meanHotLoadCount(4)
REAL*4 meanHotLoadCntnDiode(4)
REAL*4 meanColdSkyCount(4)
REAL*4 meanColdSkyCntnDiode(4)
REAL*4 diodeExcessTemp(4)
REAL*4 gain(2,4)
REAL*4 offset(2,4)
REAL*4 nonLinearGain(4)
INTEGER*2 calibrationQCflag
INTEGER*2 diodeFlag
REAL*4 receiverTemp(4)
REAL*4 receiverGain(4)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S2_NAV2/
INTEGER*2 SCE_SELECTION
INTEGER*2 SCE_RATE
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S2_NEDTINFO/
  REAL*4 NEDTinfo(4)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S2_SAMPLEHEADER/
  BYTE blanking
  INTEGER*2 earthViewFirstSample
  INTEGER*2 sampleNumber(4,4)
  INTEGER*4 tachSeconds(32)
  INTEGER*2 tachSubSeconds(32)
  INTEGER*4 indexPulseSeconds
  INTEGER*2 indexPulseSubSeconds
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S2_SCANSTATUS/
  BYTE dataQuality
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S2/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(221)
  REAL*4 Longitude(221)
  RECORD /L1BASEGMIRSS_S2_SCANSTATUS/ scanStatus
  RECORD /L1BASEGMIRSS_S2_SAMPLEHEADER/ sampleHeader
  RECORD /L1BASEGMIRSS_S2_NEDTINFO/ NEDTinfo
  RECORD /NAVIGATION/ navigation
  RECORD /L1BASEGMIRSS_S2_NAV2/ nav2
  RECORD /L1BASEGMIRSS_S2_CALIBRATION/ calibration
  RECORD /L1BASEGMIRSS_S2_CAL2/ cal2
  REAL*4 moonVectorInstFrame(3)
RECORD /L1BASEGMIRSS_S2_CALCOUNTS/ calCounts
RECORD /L1BASEGMIRSS_S2_SUNDATA/ sunData
REAL*4 incidenceAngle(221)
REAL*4 satAzimuthAngle(221)
REAL*4 solarZenAngle(221)
REAL*4 solarAzimuthAngle(221)
REAL*4 sunGlintAngle(221)
REAL*4 magneticFieldVector(3)
REAL*4 TAMmagneticFieldVector(3)
INTEGER*2 earthViewCounts(4,221)
REAL*4 Tb(4,221)
INTEGER*2 RFIFlag(2,221)

END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S1_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
  REAL*4 sunVectorInBodyFrame(3)

END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S1_CALCOUNTS/
  INTEGER*2 hotLoadReading(65,9)
  INTEGER*2 coldLoadReading(85,9)
  INTEGER*2 hotLoadnDiodeReading(65,9)
  INTEGER*2 coldLoadnDiodeReading(85,9)
  INTEGER*2 hotLoadThermisterCount(11)

END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S1_CAL2/
  INTEGER*2 trayTemperatureCount
  REAL*4 trayTemperature
  INTEGER*2 moonIndex(9)
  REAL*4 noiseDiodeTemp(6)
  INTEGER*2 TEMP_CALRES_2
  INTEGER*2 TEMP_CALRES_1
  INTEGER*2 RS_CALRES_2
  INTEGER*2 RS_CALRES_1
  INTEGER*2 BATC_CALRES_2
INTEGER*2 BATC_CALRES_1
INTEGER*2 NDIODE_MODE
INTEGER*2 RSST_NDIODE_ST
INTEGER*2 NDIODE10GHZNUM
REAL*4 hotLoadThermisterTemp(11,9)
INTEGER*2 TEMP_CALRES_4
INTEGER*2 TEMP_CALRES_5
INTEGER*2 TEMP_CALRES_6
REAL*4 TEMP_89GHZ_LO
REAL*4 TEMP_166GHZ_LO
REAL*4 TEMP_183GHZ_LO
REAL*4 TEMP_V89GHZMXR
REAL*4 TEMP_H89GHZMXR
REAL*4 TEMP_V166GHZMXR
REAL*4 TEMP_H166GHZMXR
REAL*4 TEMP_183GHZMXR
REAL*4 TEMP_RS_MR1
REAL*4 TEMP_RS_MR2
REAL*4 MR_ICA_TEMP
REAL*4 MR_LR_LEFT_TEMP
REAL*4 MR_LR_RGHT_TEMP
REAL*4 MR_LR_LOWR_TEMP
REAL*4 CSR_TEMP1
REAL*4 CSR_TEMP2
REAL*4 onOrbitDiodeExcessTemp(9)
INTEGER*2 WarmIntrusionToColdViewIndex(9,85)

END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S1_CALIBRATION/
REAL*4 hotLoadTemp(9)
REAL*4 coldSkyTemp(9)
REAL*4 onOrbitNonLinearity(9)
REAL*4 derivedNonLinearity(9)
REAL*4 meanHotLoadCount(9)
REAL*4 meanHotLoadCntnDiode(9)
REAL*4 meanColdSkyCount(9)
REAL*4 meanColdSkyCntnDiode(9)
REAL*4 diodeExcessTemp(9)
REAL*4 gain(2,9)
REAL*4 offset(2,9)
REAL*4 nonLinearGain(9)
INTEGER*2 calibrationQCflag
INTEGER*2 diodeFlag
REAL*4 receiverTemp(9)
REAL*4 receiverGain(9)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S1_NAV2/
  INTEGER*2 SCE_SELECTION
  INTEGER*2 SCE_RATE
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
  REAL*4 scAlt
  REAL*4 drpAlt
  REAL*4 scAttRollGeoc
  REAL*4 scAttPitchGeoc
  REAL*4 scAttYawGeoc
  REAL*4 scAttRollGeod
  REAL*4 scAttPitchGeod
  REAL*4 scAttYawGeod
  REAL*4 greenHourAng
  REAL*8 timeMidScan
  REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S1_NEDTINFO/
  REAL*4 NEDTinfo(9)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S1_SAMPLEHEADER/
  BYTE blanking
  INTEGER*2 earthViewFirstSample
  INTEGER*2 sampleNumber(4,9)
  INTEGER*4 tachSeconds(32)
  INTEGER*2 tachSubSeconds(32)
  INTEGER*4 indexPulseSeconds
  INTEGER*2 indexPulseSubSeconds
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S1_SCANSTATUS/
  BYTE dataQuality
BYTE missing
BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /SCANTIME/>
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay

END STRUCTURE

STRUCTURE /L1BASEGMIRSS_S1/>
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(221)
  REAL*4 Longitude(221)
  RECORD /L1BASEGMIRSS_S1_SCANSTATUS/ scanStatus
  RECORD /L1BASEGMIRSS_S1_SAMPLEHEADER/ sampleHeader
  RECORD /L1BASEGMIRSS_S1_NEDTINFO/ NEDTinfo
  RECORD /NAVIGATION/ navigation
  RECORD /L1BASEGMIRSS_S1_NAV2/ nav2
  RECORD /L1BASEGMIRSS_S1_CALIBRATION/ calibration
  RECORD /L1BASEGMIRSS_S1_CAL2/ cal2
  REAL*4 moonVectorInstFrame(3)
  RECORD /L1BASEGMIRSS_S1_CALCOUNTS/ calCounts
  RECORD /L1BASEGMIRSS_S1_SUNDATA/ sunData
  REAL*4 incidenceAngle(221)
  REAL*4 satAzimuthAngle(221)
  REAL*4 solarZenAngle(221)
  REAL*4 solarAzimuthAngle(221)
  REAL*4 sunGlintAngle(221)
REAL*4 magneticFieldVector(3)
REAL*4 TAMmagneticFieldVector(3)
ingTEGER*2 earthViewCounts(9,221)
REAL*4 Tb(9,221)
ingTEGER*2 RFIFlag(5,221)
END STRUCTURE

STRUCTURE /L1BASEGMIRSS_SWATHS/
  RECORD /L1BASEGMIRSS_S1/ S1;
  RECORD /L1BASEGMIRSS_S2/ S2;
  RECORD /L1BASEGMIRSS_S3/ S3;
  RECORD /L1BASEGMIRSS_S4/ S4;
END STRUCTURE

5.7 1BASEGMIXCAL - GMI Antenna Temperatures

The GMI BASE Product, 1BASEGMIXCAL, "GMI Antenna Temperatures," is written as a multi-Swath Structure. Swath S1 has channels 1-9: 10V 10H 19V 19H 23V 37V 37H 89V 89H. Swath S2 has channels 10-13: 166V 166H 183+/−3V 183+/−8V. S3 S4 are full rotation versions of S1 S2. 1BASEGMIXCAL is like 1BASEGMI but has overlap of 200 scans on each end of S3 and S4. The following sections describe the structure and contents of the format.

Dimension definitions:
nscan: Number of scans in the granule.
nchan1: Number of channels in Swath 1.
nchan2: Number of channels in Swath 2.
nfreq1: Number of frequencies in Swath 1.
nfreq2: Number of frequencies in Swath 2.
npix1: Number of pixels in Swath 1.
npix2: Number of pixels in Swath 2.
npix3: Number of pixels in Swath 3.
npix4: Number of pixels in Swath 4.
ncolds1: Maximum number of cold samples in Swath 1.
ncolds2: Maximum number of cold samples in Swath 2.
hots1: Maximum number of hot samples in Swath 1.
hots2: Maximum number of hot samples in Swath 2.
ntherm: Number of hot load thermisters.
LNL: Linear and non-linear.
nsamt: Number of sample types. The types are: total science GSDR, earth-view, hot load, cold sky.
ntach: Number of tachometer readings.
GMIxyz: x, y, z components in GMI instrument coordinate system.
nndiode: Number of noise diodes.
n7: Number seven.

Figure 186 through Figure 218 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 187: Data Format Structure for 1BASEGMIXCAL, S1
Figure 188: Data Format Structure for 1BASEGMIXCAL, S2
Figure 189: Data Format Structure for 1BASEGMIXCAL, S3

Figure 190: Data Format Structure for 1BASEGMIXCAL, S4

Figure 191: Data Format Structure for 1BASEGMIXCAL, S1, ScanTime
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

- scanStatus:
  - dataQuality: 1 byte, Array: nscan
  - missing: 1 byte, Array: nscan
  - modeStatus: 1 byte, Array: nscan
  - geoError: 2 bytes, Array: nscan
  - geoWarning: 2 bytes, Array: nscan
  - SCorientation: 2 bytes, Array: nscan
  - pointingStatus: 2 bytes, Array: nscan
  - acsModeMidScan: 1 byte, Array: nscan
  - targetSelectionMidScan: 1 byte, Array: nscan
  - operationalMode: 1 byte, Array: nscan
  - FractionalGranuleNumber: 8 bytes, Array: nscan

Figure 192: Data Format Structure for 1BASEGMIXCAL, S1, scanStatus

- sampleHeader:
  - blanking: 1 byte, Array: nscan
  - earthViewFirstSample: 2 bytes, Array: nscan
  - sampleNumber: 2 bytes, Array: nsamt x nchan1 x nscan
  - tachSeconds: 4 bytes, Array: ntach x nscan
  - tachSubSeconds: 2 bytes, Array: ntach x nscan
  - indexPulseSeconds: 4 bytes, Array: nscan
  - indexPulseSubSeconds: 2 bytes, Array: nscan

Figure 193: Data Format Structure for 1BASEGMIXCAL, S1, sampleHeader

- NEDTinfo:
  - NEDTinfo: 4 bytes, Array: nchan1 x nscan

Figure 194: Data Format Structure for 1BASEGMIXCAL, S1, NEDTinfo
Figure 195: Data Format Structure for 1BASEGMIXCAL, S1, navigation

Figure 196: Data Format Structure for 1BASEGMIXCAL, S1, nav2
Figure 197: Data Format Structure for 1BASEGMIXCAL, S1, calibration
Figure 198: Data Format Structure for 1BASEGMIXCAL, cal2
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

continued from last figure

![Diagram of data format structure for 1BASEGMIXCAL, S1, cal2]

Figure 199: Data Format Structure for 1BASEGMIXCAL, S1, cal2

![Diagram of data format structure for 1BASEGMIXCAL, S1, calCounts]

Figure 200: Data Format Structure for 1BASEGMIXCAL, S1, calCounts
Figure 201: Data Format Structure for 1BASEGMIXCAL, S1, sunData

Figure 202: Data Format Structure for 1BASEGMIXCAL, S2, ScanTime
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

**Figure 203: Data Format Structure for 1BASEGMIXCAL, S2, scanStatus**

- dataQuality: 1 byte, Array: nscan
- missing: 1 byte, Array: nscan
- modeStatus: 1 byte, Array: nscan
- geoError: 2 bytes, Array: nscan
- geoWarning: 2 bytes, Array: nscan
- SCorientation: 2 bytes, Array: nscan
- pointingStatus: 2 bytes, Array: nscan
- acsModeMidScan: 1 byte, Array: nscan
- targetSelectionMidScan: 1 byte, Array: nscan
- operationalMode: 1 byte, Array: nscan
- FractionalGranuleNumber: 8 bytes, Array: nscan

**Figure 204: Data Format Structure for 1BASEGMIXCAL, S2, sampleHeader**

- blanking: 1 byte, Array: nscan
- earthViewFirstSample: 2 bytes, Array: nscan
- sampleNumber: 2 bytes, Array: nsamt x nchan2 x nscan
- tachSeconds: 4 bytes, Array: ntach x nscan
- tachSubSeconds: 2 bytes, Array: ntach x nscan
- indexPulseSeconds: 4 bytes, Array: nscan
- indexPulseSubSeconds: 2 bytes, Array: nscan

**Figure 205: Data Format Structure for 1BASEGMIXCAL, S2, NEDTinfo**

- NEDTinfo: 4 bytes, Array: nchan2 x nscan
Figure 206: Data Format Structure for 1BASEGMIXCAL, S2, navigation

Figure 207: Data Format Structure for 1BASEGMIXCAL, S2, nav2
Figure 208: Data Format Structure for 1BASEGMIXCAL, S2, calibration
Figure 209: Data Format Structure for 1BASEGMIXCAL, cal2
continued from last figure

- TEMP_183GHZMXR 4 bytes Array: nscan
- TEMP_RS_MRI 4 bytes Array: nscan
- TEMP_RS_MRII 4 bytes Array: nscan
- MR_ICA_TEMP 4 bytes Array: nscan
- MR_LR_LEFT_TEMP 4 bytes Array: nscan
- MR_LR_RIGHT_TEMP 4 bytes Array: nscan
- MR_LR_LOW_RIGHT_TEMP 4 bytes Array: nscan
- CSR_TEMP1 4 bytes Array: nscan
- CSR_TEMP2 4 bytes Array: nscan
- onOrbitDiodeExcessTemp 4 bytes Array: nchan2 x nscan
- WarmIntrusionToColdViewIndex 2 bytes Array: nchan2 x ncholds2 x nscan

Figure 210: Data Format Structure for 1BASEGMIXCAL, S2, cal2

- hotLoadReading 2 bytes Array: nhots2 x nchan2 x nscan
- coldLoadReading 2 bytes Array: ncholds2 x nchan2 x nscan
- hotLoadnDiodeReading 2 bytes Array: nhots2 x nchan2 x nscan
- coldLoadnDiodeReading 2 bytes Array: ncholds2 x nchan2 x nscan
- hotLoadThermisterCount 2 bytes Array: ntherm x nscan

Figure 211: Data Format Structure for 1BASEGMIXCAL, S2, calCounts
5 STANDARD GPM PRODUCTS

![Diagram of data format structure for 1BASEGMIXCAL, S2, sunData](image)

Figure 212: Data Format Structure for 1BASEGMIXCAL, S2, sunData

![Diagram of data format structure for 1BASEGMIXCAL, S3, ScanTime](image)

Figure 213: Data Format Structure for 1BASEGMIXCAL, S3, ScanTime

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

Figure 214: Data Format Structure for 1BASEGMIXCAL, S3, scanStatus

Figure 215: Data Format Structure for 1BASEGMIXCAL, S3, calibration
Figure 216: Data Format Structure for IBASEGMIXCAL, S4, ScanTime

Figure 217: Data Format Structure for IBASEGMIXCAL, S4, scanStatus
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

Figure 218: Data Format Structure for 1BASEGMIXCAL, S4, calibration
all data products. See Metadata for GPM Products for details.

S1 (Swath)

S1_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S1)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npix1 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npix1 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S1)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.

modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit =
1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other
bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorient not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
5.7 BASEGMIXCAL - GMI Antenna Temperatures

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

Value Meaning
0    +X forward (yaw 0)
180  -X forward (yaw 180)
-8000 Non-nominal pointing
-9999 Missing

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

Value Meaning
Nominal pointing in Mission Science Mode
1 GPS point solution stale and PVT ephemeris used
2 GEONS solution stale and GEONS ephemeris used
-8000 Non-nominal mission science orientation
-9999 Missing

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**sampleHeader** (Group in S1)

**blanking** (1-byte integer, array size: nscan):

Value of 0 = Blanking off
Value of 1 = Blanking on

**earthViewFirstSample** (2-byte integer, array size: nscan):
Sample number of the first earth view. Values range from 0 to 512. Special values are defined as:
-9999 Missing value

**sampleNumber** (2-byte integer, array size: nsamt x nchan1 x nscan):
Number of valid samples in scan. Values range from 0 to 512. Special values are defined as:
-9999 Missing value

**tachSeconds** (4-byte unsigned integer, array size: ntach x nscan):
Tachometer seconds. Values are in second. Special values are defined as:
0 Missing value

**tachSubSeconds** (2-byte unsigned integer, array size: ntach x nscan):
Tachometer sub seconds. Values range from 0 to 62499 in units of 16 microseconds. The missing value is 65535.

**indexPulseSeconds** (4-byte unsigned integer, array size: nscan):
Index Pulse seconds. Values are in second. Special values are defined as:
0 Missing value

**indexPulseSubSeconds** (2-byte unsigned integer, array size: nscan):
Index Pulse subseconds. Values range from 0 to 62499 in units of 16 microseconds. The missing value is 65535.

**NEDTinfo** (Group in S1)

**NEDTinfo** (4-byte float, array size: nchan1 x nscan):
NEDT (Noise Equivalent Differential Temperature) for each channel.

navigation (Group in S1)

scPos (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector (ms$^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity
direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that
pitch and roll will have twice orbital frequency components due to the onboard control
system following the oblate geodetic Earth horizon. Note also that the yaw value will
show an orbital frequency component relative to the Earth fixed ground track due to the
Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees.
Special values are defined as:
-9999.9  Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time.
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values
range from -135 to 225 degrees. Special values are defined as:
-9999.9  Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order
of the components in the file is roll, pitch, and yaw. However, the angles are computed
using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll
for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic
Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the
Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity
opposite the orbit normal direction, and the X-axis is approximately in the velocity
direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
-9999.9  Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values
range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values
range from -135 to 225 degrees. Special values are defined as:
-9999.9  Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coor

## BASEGMIXCAL - GMI Antenna Temperatures

- The rotation angle from Geocentric Inertial Coordinates to Earth Fixed Coordinates.
- The rotation angle is expressed in degrees. Values range from 0 to 360 degrees.

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980.
TimeMidScan is used as the reference time for the scPos and scVel values. Values range
STANDARD GPM PRODUCTS

from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**nav2** (Group in S1)

**SCE_SELECTION** (2-byte unsigned integer, array size: nscan):
The current SCE selection setting. Special values are defined as:
0 Missing value

**SCE_RATE** (2-byte unsigned integer, array size: nscan):
The SMA rotational rate reported by the SCE. To obtain the spin rate in RPM, multiply SCE_RATE by 0.002999106 Values range from 1 to 65535 count. Special values are defined as:
0 Missing value

**calibration** (Group in S1)

**hotLoadTemp** (4-byte float, array size: nchan1 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray temperature and averaged over closest 5 scans Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**coldSkyTemp** (4-byte float, array size: nchan1 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**onOrbitNonLinearity** (4-byte float, array size: nchan1 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table . Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**derivedNonLinearity** (4-byte float, array size: nchan1 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special
values are defined as:
  -9999.9 Missing value

**meanHotLoadCount** (4-byte float, array size: nchan1 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
  -9999.9 Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Hot Load Plus Noise Diode counts. Averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
  -9999.9 Missing value

**meanColdSkyCount** (4-byte float, array size: nchan1 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
  -9999.9 Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
  -9999.9 Missing value

**diodeExcessTemp** (4-byte float, array size: nchan1 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature=diodeExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature=diodeExcessTemp + hotLoad-Temp. Values range from 0 to 400 K. Special values are defined as:
  -9999.9 Missing value

**gain** (4-byte float, array size: LNL x nchan1 x nscan):
gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
  -9999.9 Missing value

**offset** (4-byte float, array size: LNL x nchan1 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan1 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
  -9999.9 Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
calibrationQCflag. value 0 indicates good calibration. Values range from 0 to 15. Special values are defined as:
  -9999 Missing value

**diodeFlag** (2-byte integer, array size: nscan):
Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

**receiverTemp** (4-byte float, array size: nchan1 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**receiverGain** (4-byte float, array size: nchan1 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9 Missing value

**cal2** (Group in S1)

**trayTemperatureCount** (2-byte unsigned integer, array size: nscan):
Counts to derive hot load tray temperature. Values range from 0 to 65535 count. Special values are defined as:
0 Missing value

**trayTemperature** (4-byte float, array size: nscan):
Derive hot load tray temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**moonIndex** (2-byte unsigned integer, array size: nchan1 x nscan):
Index determined by the angle between moon vector and cold sample vectors. 0 means angles between moon vector and all cold view vectors are greater than 5 degrees. Non-zero value means the number of cold samples that may be contaminated by moon. Values range from 0 to 100. Special values are defined as:
0 Missing value

**noiseDiodeTemp** (4-byte float, array size: nndiode x nscan):
Physical temperature of noise diode. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**TEMP_CALRES_2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for PRT temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0 Missing value

**TEMP_CALRES_1** (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for PRT temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
0 Missing value
**RS_CALRES.2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
- 0  Missing value

**RS_CALRES.1** (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for tray temperature and receiver temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
- 0  Missing value

**BATC_CALRES.2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
- 0  Missing value

**BATC_CALRES.1** (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for noise diode temperature retrieval. Values range from 0 to 65535 count. Special values are defined as:
- 0  Missing value

**NDIODE_MODE** (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Mode. 0 = On every scan, 1 = On every other scan, 2 = Off. Values range from 0 to 2 count. Special values are defined as:
- 65535  Missing value

**RSST_NDIODE_ST** (2-byte unsigned integer, array size: nscan):
Noise diode state during the scan. 0 = Noise diodes OFF for the scan, 1 = Noise diodes ON for the scan. Values range from 0 to 1 count. Special values are defined as:
- 65535  Missing value

**NDIODE10GHZNUM** (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Start Sample Number, i.e., the sample number where noise diodes are turned on. Values range from 0 to 500 count. Special values are defined as:
- 65535  Missing value

**hotLoadThermisterTemp** (4-byte float, array size: ntherm x nchan1 x nscan):
Hot Load Thermister Temperature of 11 PRTs. Values range from 0 to 400 K. Special values are defined as:
- -9999.9  Missing value

**TEMP_CALRES.4** (2-byte unsigned integer, array size: nscan):
Low calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
- 0  Missing value

**TEMP_CALRES.5** (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
- 0  Missing value
TEMP\_CALRES\_6 (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
- 0  Missing value

TEMP\_89GHZ\_LO (4-byte float, array size: nscan):
89 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP\_166GHZ\_LO (4-byte float, array size: nscan):
166 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP\_183GHZ\_LO (4-byte float, array size: nscan):
183 GHZ Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP\_V89GHZMXR (4-byte float, array size: nscan):
89 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP\_H89GHZMXR (4-byte float, array size: nscan):
89 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP\_V166GHZMXR (4-byte float, array size: nscan):
166 GHZ V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP\_H166GHZMXR (4-byte float, array size: nscan):
166 GHZ H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP\_183GHZMXR (4-byte float, array size: nscan):
183 GHZ Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value

TEMP\_RS\_MR1 (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 1 Values range from 0 to 500 K. Special values are defined as:
-9999.9  Missing value
TEMP_RS_MR2 (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 2 Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

MR_ICA_TEMP (4-byte float, array size: nscan):
Main Reflector Temperature Read By ICA Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

MR_LR_LEFT_TEMP (4-byte float, array size: nscan):
Main Reflector left Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

MR_LR_RGHT_TEMP (4-byte float, array size: nscan):
Main Reflector right Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

MR_LR_LOWR_TEMP (4-byte float, array size: nscan):
Main Reflector lower Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

CSR_TEMP1 (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 1 Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

CSR_TEMP2 (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 2 Values range from 0 to 500 K. Special values are defined as:
-9999.9 Missing value

onOrbitDiodeExcessTemp (4-byte float, array size: nchan1 x nscan):
Diode Excess Temperature derived from on orbit trended look-up tables as a function of noise diode temperature from telemetry. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

WarmIntrusionToColdViewIndex (2-byte unsigned integer, array size: nchan1 x ncolds1 x nscan):
Index flag to determine if a cold view sample is contaminated by certain warmer sources. If the value is 0, the sample is good and the count is used in calibration. If the value is non-zero, the sample is contaminated and excluded in calibration.

0: Good sample
1: Bad sample determined by limit check
2: Bad sample determined by 2D medium filter

Values range from 0 to 2. Special values are defined as:
   65535  Missing value

**moonVectorInstFrame** (4-byte float, array size: GMIxyz x nscan):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
   -9999.9  Missing value

**calCounts** (Group in S1)

**hotLoadReading** (2-byte unsigned integer, array size: nhots1 x nchan1 x nscan):
Hot Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

**coldLoadReading** (2-byte unsigned integer, array size: ncolds1 x nchan1 x nscan):
Cold Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

**hotLoadnDiodeReading** (2-byte unsigned integer, array size: nhots1 x nchan1 x nscan):
Hot Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

**coldLoadnDiodeReading** (2-byte unsigned integer, array size: ncolds1 x nchan1 x nscan):
Cold Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

**hotLoadThermisterCount** (2-byte unsigned integer, array size: ntherm x nscan):
Counts from 11 PRTs in the hot load. Values range from 0 to 65534 counts. Special values are defined as:
   65535  Missing value

**sunData** (Group in S1)

**solarBetaAngle** (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
   -9999.9  Missing value
phaseFromOrbitMidnight (4-byte float, array size: nscan):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of
the Earth center from the spacecraft and positive toward the spacecraft velocity direction
so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees
occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local
orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator.
Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan):
The separation angle between the Sun and Earth directions from the spacecraft. Values
range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan):
The angle between the center of the Earth and the horizon edge. The sun is above
the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius.
Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow,
based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from
0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nscan):
The instantaneous angular rate of the spacecraft around the orbit. Values range from
0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values
range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such
that +Z is nominally toward the Earth and gives the instrument spin axis, and data is
collected nominally centered about the +X direction. Values range from 0 to 1.0. Special
values are defined as:
-9999.9 Missing value

incidenceAngle (4-byte float, array size: npix1 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel
location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value
satAzimuthAngle (4-byte float, array size: npix1 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction
to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarZenAngle (4-byte float, array size: npix1 x nscan):
The angle between the local pixel geodetic zenith and the direction to the sun. Values
range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarAzimuthAngle (4-byte float, array size: npix1 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction
to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (4-byte float, array size: npix1 x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off
the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location
on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector
from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s
surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle
is the angular separation between the Reflected Satellite View Vector and the Sun Vector.
When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like)
sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

magneticFieldVector (4-byte float, array size: GMIxyz x nscan):
Magnetometer volt reading in TAM (x, y, z) coordinate system. Used to perform along-
scan correction of earth view counts. (The TAM (x,y,z) coordinate system is similar to
GPM S/C coordinate system but y and z axis are rotated by 180 degrees.) Values range
from -500 to 500 V. Special values are defined as:
-9999.9 Missing value

TAMmagneticFieldVector (4-byte float, array size: GMIxyz x nscan):
Magnetic Field derived from GPM three-axis magnetometer (TAM). Values range from
-1000 to 1000 V. Special values are defined as:
-9999.9 Missing value

earthViewCounts (2-byte unsigned integer, array size: nchan1 x npix1 x nscan):
Earth view counts. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value

Tb (4-byte float, array size: nchan1 x npix1 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined
as:
-9999.9 Missing value

RFIFlag (2-byte integer, array size: nfreq1 x npix1 x nscan):
Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: Not affected by RFI.
1: Affected by RFI with X-cal filter.
2: Affected by RFI with RSS filter.
3-7: Spare
-9999: Missing

S2 (Swath)

S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value
MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npix2 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npix2 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S2)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
</tbody>
</table>
4 Housekeeping (HK) telemetry packet missing
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^{**i}$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^{**i}$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
</tbody>
</table>
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
</tbody>
</table>
pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is
good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit
in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
Status of the GMI instrument.
Bit Meaning if bit = 1
0 Receiver status (0=ON, 1=OFF)
1 Spinup Status (0=ON, 1=OFF)

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**sampleHeader** (Group in S2)

**blanking** (1-byte integer, array size: nscan):

Value of 0 = Blanking off
Value of 1 = Blanking on

**earthViewFirstSample** (2-byte integer, array size: nscan):
Sample number of the first earth view. Values range from 0 to 512. Special values are defined as:
-9999 Missing value

**sampleNumber** (2-byte integer, array size: nsamt x nchan2 x nscan):
Number of valid samples in scan. Values range from 0 to 512. Special values are defined as:
-9999 Missing value

**tachSeconds** (4-byte unsigned integer, array size: ntach x nscan):
Tachometer seconds. Values are in second. Special values are defined as:
0 Missing value

**tachSubSeconds** (2-byte unsigned integer, array size: ntach x nscan):
Tachometer sub-seconds. Values range from 0 to 62499 in units of 16 microseconds. The missing value is 65535.

**indexPulseSeconds** (4-byte unsigned integer, array size: nscan):
Index Pulse seconds. Values are in second. Special values are defined as:
0 Missing value

**indexPulseSubSeconds** (2-byte unsigned integer, array size: nscan):
Index Pulse sub-seconds. Values range from 0 to 62499 in units of 16 microseconds. The missing value is 65535.
NEDTinfo (Group in S2)

NEDTinfo (4-byte float, array size: nchan2 x nsan):

NEDT (Noise Equivalent Differential Temperature) for each channel.

navigation (Group in S2)

scPos (4-byte float, array size: XYZ x nsan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coor-
dinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan
period). Values range from -10000000 to 10000000 m. Special values are defined as:
  -9999.9  Missing value

scVel (4-byte float, array size: XYZ x nsan):
The velocity vector (ms$^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time.
Values range from -10000000 to 10000000 m/s. Special values are defined as:
  -9999.9  Missing value

scLat (4-byte float, array size: nsan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values
range from -70 to 70 degrees. Special values are defined as:
  -9999.9  Missing value

scLon (4-byte float, array size: nsan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values
range from -180 to 180 degrees. Special values are defined as:
  -9999.9  Missing value

scAlt (4-byte float, array size: nsan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values
range from 350000 to 500000 m. Special values are defined as:
  -9999.9  Missing value

dprAlt (4-byte float, array size: nsan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from
DPR science telemetry. This is empty in non-DPR products. Values range from 350000
to 500000 m. Special values are defined as:
  -9999.9  Missing value

scAttRollGeo (4-byte float, array size: nsan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The or-
der of the components in the file is roll, pitch, and yaw. However, the angles are computed
using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

\textit{scAttPitchGeoc} (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

\textit{scAttYawGeoc} (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

\textit{scAttRollGeod} (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

\textit{scAttPitchGeod} (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

\textit{scAttYawGeod} (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

\textit{greenHourAng} (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coor-
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980.
timeMidScan is used as the reference time for the scPos and scVel values. Values range
from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0
to 100 s. Special values are defined as:
-9999.9 Missing value

nav2 (Group in S2)

SCE_SELECTION (2-byte unsigned integer, array size: nscan):
The current SCE selection setting. Special values are defined as:
  0 Missing value

SCE_RATE (2-byte unsigned integer, array size: nscan):
The SMA rotational rate reported by the SCE. To obtain the spin rate in RPM, multiply
SCE_RATE by 0.002999106 Values range from 1 to 65535 count. Special values are defined
as:
  0 Missing value

calibration (Group in S2)

hotLoadTemp (4-byte float, array size: nchan2 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load.
For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89
GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray
temperature and averaged over closest 5 scans Values range from 0 to 400 K. Special
values are defined as:
-9999.9 Missing value

coldSkyTemp (4-byte float, array size: nchan2 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined
as:
-9999.9 Missing value

onOrbitNonLinearity (4-byte float, array size: nchan2 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table. Values
range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**derivedNonLinearity** (4-byte float, array size: nchan2 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**meanHotLoadCount** (4-byte float, array size: nchan2 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9  Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Hot Load Plus Noise Diode counts Averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9  Missing value

**meanColdSkyCount** (4-byte float, array size: nchan2 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9  Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**diodeExcessTemp** (4-byte float, array size: nchan2 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature=diodeExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature=diodeExcessTemp + hotLoadTemp. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**gain** (4-byte float, array size: LNL x nchan2 x nscan):
gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9  Missing value

**offset** (4-byte float, array size: LNL x nchan2 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan2 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9  Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
calibrationQCflag. value 0 indicates good calibration. Values range from 0 to 15. Special
values are defined as:
-9999   Missing value

**diodeFlag** (2-byte integer, array size: nscan):

Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

**receiverTemp** (4-byte float, array size: nchan2 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9   Missing value

**receiverGain** (4-byte float, array size: nchan2 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9   Missing value

**cal2** (Group in S2)

**trayTemperatureCount** (2-byte unsigned integer, array size: nscan):
Counts to derive hot load tray temperature. Values range from 0 to 65534 count. Special values are defined as:
65535   Missing value

**trayTemperature** (4-byte float, array size: nscan):
Derive hot load tray temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9   Missing value

**moonIndex** (2-byte unsigned integer, array size: nchan2 x nscan):
Index determined by the angle between moon vector and cold sample vectors. 0 means angles between moon vector and all cold view vectors are greater than 5 degrees. Non-zero value means the number of cold samples that may be contaminated by moon. Values range from 0 to 100. Special values are defined as:
0   Missing value

**noiseDiodeTemp** (4-byte float, array size: ndiode x nscan):
Physical temperature of noise diode. Values range from 0 to 400 K. Special values are defined as:
-9999.9   Missing value

**TEMP_CALRES.2** (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for PRT temperature retrieval. Values range from
0 to 65535 count. Special values are defined as:
0  Missing value

TEMP_CALRES.1 (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for PRT temperature retrieval. Values range from
0 to 65535 count. Special values are defined as:
0  Missing value

RS_CALRES.2 (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for tray temperature and receiver temperature
retrieval. Values range from 0 to 65535 count. Special values are defined as:
0  Missing value

BATC_CALRES.1 (2-byte unsigned integer, array size: nscan):
Count of low calibration resistor used for tray temperature and receiver temperature
retrieval. Values range from 0 to 65535 count. Special values are defined as:
0  Missing value

BATC_CALRES.2 (2-byte unsigned integer, array size: nscan):
Count of high calibration resistor used for noise diode temperature retrieval. Values range
from 0 to 65535 count. Special values are defined as:
0  Missing value

NDIODE_MODE (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Mode. 0 = On every scan, 1 = On every other scan, 2
= Off. Values range from 0 to 2 count. Special values are defined as:
65535  Missing value

RSST_NDIODE_ST (2-byte unsigned integer, array size: nscan):
Noise diode state during the scan. 0 = Noise diodes OFF for the scan, 1 = Noise diodes
ON for the scan. Values range from 0 to 1 count. Special values are defined as:
65535  Missing value

NDIODE10GHZNUM (2-byte unsigned integer, array size: nscan):
RS configuration of Noise Diode Start Sample Number, i.e., the sample number where
noise diodes are turned on. Values range from 0 to 500 count. Special values are defined
as:
65535  Missing value

hotLoadThermisterTemp (4-byte float, array size: ntherm x nchan2 x nscan):
Hot Load Thermister Temperature of 11 PRTs. Values range from 0 to 400 K. Special
values are defined as:
-9999.9  Missing value
TEMP_CALRES.4 (2-byte unsigned integer, array size: nscan):
Low calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
   0 Missing value

TEMP_CALRES.5 (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
   0 Missing value

TEMP_CALRES.6 (2-byte unsigned integer, array size: nscan):
High calibration resistor count. Values range from 0 to 65535 count. Special values are defined as:
   0 Missing value

TEMP_89GHZ_LO (4-byte float, array size: nscan):
89 GHz Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_166GHZ_LO (4-byte float, array size: nscan):
166 GHz Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_183GHZ_LO (4-byte float, array size: nscan):
183 GHz Local Oscillator Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_V89GHZMXR (4-byte float, array size: nscan):
89 GHz V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_H89GHZMXR (4-byte float, array size: nscan):
89 GHz H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_V166GHZMXR (4-byte float, array size: nscan):
166 GHz V channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_H166GHZMXR (4-byte float, array size: nscan):
166 GHz H channel Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value
TEMP_183GHZMXR (4-byte float, array size: nscan):
183 GHZ Mixer PRE-AMP Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_RS_MR1 (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 1 Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

TEMP_RS_MR2 (4-byte float, array size: nscan):
Main Reflector Temperature Read By RS 2 Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

MR_ICA_TEMP (4-byte float, array size: nscan):
Main Reflector Temperature Read By ICA Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

MR_LR_LEFT_TEMP (4-byte float, array size: nscan):
Main Reflector left Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

MR_LR_RGHT_TEMP (4-byte float, array size: nscan):
Main Reflector right Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

MR_LR_LOWR_TEMP (4-byte float, array size: nscan):
Main Reflector lower Launch Restraint Temperature Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

CSR_TEMP1 (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 1 Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

CSR_TEMP2 (4-byte float, array size: nscan):
Cold Sky Reflector Temperature 2 Values range from 0 to 500 K. Special values are defined as:
   -9999.9 Missing value

onOrbitDiodeExcessTemp (4-byte float, array size: nchan2 x nscan):
Diode Excess Temperature derived from on orbit trended look-up tables as a function of noise diode temperature from telemetry. Values range from 0 to 400 K. Special values are defined as:
   -9999.9 Missing value
WarmIntrusionToColdViewIndex (2-byte unsigned integer, array size: nchan2 x ncolds2 x nscan):
Index flag to determine if a cold view sample is contaminated by certain warmer sources. If the value is 0, the sample is good and the count is used in calibration. If the value is non-zero, the sample is contaminated and excluded in calibration.

0: Good sample
1: Bad sample determined by limit check
2: Bad sample determined by 2D medium filter

Values range from 0 to 2. Special values are defined as:
   65535  Missing value

moonVectorInstFrame (4-byte float, array size: GMIxyz x nscan):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
   -9999.9  Missing value

calCounts (Group in S2)

hotLoadReading (2-byte unsigned integer, array size: nhots2 x nchan2 x nscan):
Hot Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

coldLoadReading (2-byte unsigned integer, array size: ncolds2 x nchan2 x nscan):
Cold Load Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

hotLoadnDiodeReading (2-byte unsigned integer, array size: nhots2 x nchan2 x nscan):
Hot Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

coldLoadnDiodeReading (2-byte unsigned integer, array size: ncolds2 x nchan2 x nscan):
Cold Load Plus Diode Reading. Values range from 0 to 65535 counts. Special values are defined as:
   0  Missing value

hotLoadThermisterCount (2-byte unsigned integer, array size: ntherm x nscan):
Counts from 11 PRTs in the hot load Values range from 0 to 65534 count. Special values are defined as:
   65535  Missing value
sunData (Group in S2)

solarBetaAngle (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nscan):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special
values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npix2 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npix2 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: npix2 x nscan):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: npix2 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npix2 x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**magneticFieldVector** (4-byte float, array size: GMIxyz x nscan):
Magnetometer volt reading in TAM (x, y, z) coordinate system. Used to perform along-scan correction of earth view counts. (The TAM (x,y,z) coordinate system is similar to GPM S/C coordinate system but y and z axis are rotated by 180 degrees.) Values range from -500 to 500 V. Special values are defined as:
-9999.9 Missing value

**TAMmagneticFieldVector** (4-byte float, array size: GMIxyz x nscan):
Magnetic Field derived from GPM three-axis magnetometer (TAM). Values range from -1000 to 1000 V. Special values are defined as:
-9999.9 Missing value

**earthViewCounts** (2-byte unsigned integer, array size: nchan2 x npix2 x nscan):
Earth view counts. Values range from 0 to 65535 counts. Special values are defined as:
0 Missing value
**Tb** (4-byte float, array size: nchan2 x npix2 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**RFIFlag** (2-byte integer, array size: nfreq2 x npix2 x nscan):
Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:
- 0: Not affected by RFI.
- 1: Affected by RFI with X-cal filter.
- 2: Affected by RFI with RSS filter.
- 3-7: Spare
- -9999: Missing

**S3** (Swath)

**S3_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S3)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- 9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
- 99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
- 99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
- 99 Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npix3 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npix3 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S3)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>
missing (1-byte integer, array size: nsan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

modeStatus (1-byte integer, array size: nsan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit =
1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other
bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nsan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).
Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
</tbody>
</table>
1. Negative scan time, invalid input
2. Error getting spacecraft attitude at scan mid-time
3. Error getting spacecraft ephemeris at scan mid-time
4. Invalid input non-unit ray vector for any pixel
5. Ray misses Earth for any pixel with normal pointing
6. Nadir calculation error for subsatellite position
7. Pixel count with geolocation error over threshold
8. Error in getting spacecraft attitude for any pixel
9. Error in getting spacecraft ephemeris for any pixel
10. Spare (always 0)
11. Spare (always 0)
12. Spare (always 0)
13. Spare (always 0)
14. Spare (always 0)
15. Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$ the unsigned integer value is $2^{**i}$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of mo-
tion, measured clockwise facing down. We define \( v \) in the same direction as the spacecraft axis \(+X\), which is also the center of the GMI scan. If \( \text{SCorientation} \) is not 0 or 180, a bit is set to 1 in \( \text{modeStatus} \).

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(+X) forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>(-X) forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: \( n\text{scan} \)):

pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in \( \text{modeStatus} \) is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**acsModeMidScan** (1-byte integer, array size: \( n\text{scan} \)):

acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: \( n\text{scan} \)):

targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, (+X) in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, (+X) in flight direction</td>
</tr>
</tbody>
</table>
operationalMode (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

calibration (Group in S3)

hotLoadTemp (4-byte float, array size: nchan1 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89 GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray temperature and averaged over closest 5 scans Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value
coldSkyTemp (4-byte float, array size: nchan1 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

onOrbitNonLinearity (4-byte float, array size: nchan1 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated lookup table . Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value
derivedNonLinearity (4-byte float, array size: nchan1 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value
**meanHotLoadCount** (4-byte float, array size: nchan1 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Hot Load Plus Noise Diode counts Averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9  Missing value

**meanColdSkyCount** (4-byte float, array size: nchan1 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan1 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9  Missing value

**diodeExcessTemp** (4-byte float, array size: nchan1 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature=diodeExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature=diodeExcessTemp + hotLoad-Temp. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**gain** (4-byte float, array size: LNL x nchan1 x nscan):
Gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9  Missing value

**offset** (4-byte float, array size: LNL x nchan1 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan1 x nscan):
The nonlinear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9  Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
Calibration QCflag. value 0 indicates good calibration. Values range from 0 to 15. Special values are defined as:
-9999  Missing value

**diodeFlag** (2-byte integer, array size: nscan):
Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

receiverTemp (4-byte float, array size: nchan1 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

receiverGain (4-byte float, array size: nchan1 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9 Missing value

incidenceAngle (4-byte float, array size: npix3 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Tb (4-byte float, array size: nchan1 x npix3 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

S4 (Swath)

S4_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S4)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value
Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousands of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npix4 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npix4 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S4)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
</tbody>
</table>
6 modeStatus is not zero

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2^i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2^i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.
Bit Meaning if bit = 1
0 Latitude limit exceeded for viewed pixel locations
1 Negative scan time, invalid input
2 Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan): A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

Bit Meaning if bit = 1
0 Ephemeris Gap Interpolated
1 Attitude Gap Interpolated
2 Attitude jump/discontinuity
3 Attitude out of range
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.
Value Meaning
0  S/C Z axis nadir, +X in flight direction
1  Flight Z axis nadir, +X in flight direction
2  S/C Z axis nadir, -X in flight direction
3  Flight Z axis nadir, -X in flight direction
4  +90 yaw for DPR antenna pattern calibration
5  -90 yaw for DPR antenna pattern calibration
-99 Missing

**operationalMode** (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**calibration** (Group in S4)

**hotLoadTemp** (4-byte float, array size: nchan2 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load.
For 10, 166, 183 GHZ channels, they are averages of PRT 1,7,8,9,10. For 18, 23, 36, 89
GHZ channels, they are averages of PRT 2,11,12,13,14. The values are corrected by tray
temperature and averaged over closest 5 scans Values range from 0 to 400 K. Special
values are defined as:
-9999.9 Missing value

**coldSkyTemp** (4-byte float, array size: nchan2 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**onOrbitNonLinearity** (4-byte float, array size: nchan2 x nscan):
The on Orbit Non-Linearity Tnl computed from ground calibrated u look-up table . Values
range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value
**derivedNonLinearity** (4-byte float, array size: nchan2 x nscan):
Non-Linearity Tnl derived from 4-point calibration. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCount** (4-byte float, array size: nchan2 x nscan):
The mean Hot Load Count. Averaged over all Hot samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanHotLoadCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Hot Load Plus Noise Diode counts. Averaged over all samples and closest 5 scans. Values range from 0 to 65535. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCount** (4-byte float, array size: nchan2 x nscan):
The mean Cold Sky Count. Averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**meanColdSkyCntnDiode** (4-byte float, array size: nchan2 x nscan):
The mean coupled Cold Sky Plus Noise Diode counts, averaged over all samples and closest 5 scans. Values range from 0 to 65535 count. Special values are defined as:
-9999.9 Missing value

**diodeExcessTemp** (4-byte float, array size: nchan2 x nscan):
The Noise Diode Excess Temperature. Cold and diode Coupled Temperature=diodeExcessTemp + coldSkyTemp. Hot and diode Coupled Temperature=diodeExcessTemp + hotLoad-Temp. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**gain** (4-byte float, array size: LNL x nchan2 x nscan):
Gain[0] determine the total Ta gain. Ta=offset[0]+gain[0]*earthCount+ nonLinearGain*earthCount*earthCount. Nonlinearity=offset[1]+gain[1]*earthCount+ nonLinearGain*earthCount*earthCount. Values range from 0 to 1 K. Special values are defined as:
-9999.9 Missing value

**offset** (4-byte float, array size: LNL x nchan2 x nscan):
Offset[0] determine the total Ta offset. min=-999 K (from -1 K), max=999 K (from 1 K). Missing value is -9999.9.

**nonLinearGain** (4-byte float, array size: nchan2 x nscan):
The non-linear gain. Values range from -1 to 1 K. Special values are defined as:
-9999.9 Missing value

**calibrationQCflag** (2-byte integer, array size: nscan):
Calibration QC flag. Value 0 indicates good calibration. Values range from 0 to 15. Special values are defined as:
-9999 Missing value
**diodeFlag** (2-byte integer, array size: nscan):

Diode flag.
0, Noise Diode on
2, Noise Diode off
5, Noise Diode status unknown

**receiverTemp** (4-byte float, array size: nchan2 x nscan):
The receiver temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**receiverGain** (4-byte float, array size: nchan2 x nscan):
The receiver gain. Values range from 0 to 100. Special values are defined as:
-9999.9  Missing value

**incidenceAngle** (4-byte float, array size: npix4 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**Tb** (4-byte float, array size: nchan2 x npix4 x nscan):
Earth view antenna temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**C Structure Header file:**

```c
#ifndef _TK_1BASEGMIXCAL_H_
define _TK_1BASEGMIXCAL_H_

#ifndef _L1BASEGMIXCAL_S4_CALIBRATION_
define _L1BASEGMIXCAL_S4_CALIBRATION_

typedef struct {
    float hotLoadTemp[4];
    float coldSkyTemp[4];
    float onOrbitNonLinearity[4];
    float derivedNonLinearity[4];
    float meanHotLoadCount[4];
    float meanHotLoadCntnDiode[4];
    float meanColdSkyCount[4];
    float meanColdSkyCntnDiode[4];
    float diodeExcessTemp[4];
    float gain[4][2];
    float offset[4][2];
```
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

    float nonLinearGain[4];
    short calibrationQCflag;
    short diodeFlag;
    float receiverTemp[4];
    float receiverGain[4];
} L1BASEGMIXCAL_S4_CALIBRATION;

#endif

#ifndef _L1BASEGMIXCAL_S4_SCANSTATUS_
#define _L1BASEGMIXCAL_S4_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMIXCAL_S4_SCANSTATUS;

#endif

#ifndef _L1BASEGMIXCAL_S4_
#define _L1BASEGMIXCAL_S4_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[500];
    float Longitude[500];
    L1BASEGMIXCAL_S4_SCANSTATUS scanStatus;
    L1BASEGMIXCAL_S4_CALIBRATION calibration;
    float incidenceAngle[500];
    float Tb[500][4];
} L1BASEGMIXCAL_S4;

#endif
#ifndef _L1BASEGMIXCAL_S3_CALIBRATION_
#define _L1BASEGMIXCAL_S3_CALIBRATION_

typedef struct {
    float hotLoadTemp[9];
    float coldSkyTemp[9];
    float onOrbitNonLinearity[9];
    float derivedNonLinearity[9];
    float meanHotLoadCount[9];
    float meanHotLoadCntnDiode[9];
    float meanColdSkyCount[9];
    float meanColdSkyCntnDiode[9];
    float diodeExcessTemp[9];
    float gain[9][2];
    float offset[9][2];
    float nonLinearGain[9];
    short calibrationQCflag;
    short diodeFlag;
    float receiverTemp[9];
    float receiverGain[9];
} L1BASEGMIXCAL_S3_CALIBRATION;

#endif

#ifndef _L1BASEGMIXCAL_S3_SCANSTATUS_
#define _L1BASEGMIXCAL_S3_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMIXCAL_S3_SCANSTATUS;

#endif
#ifndef _L1BASEGMIXCAL_S3_
define _L1BASEGMIXCAL_S3_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[500];
    float Longitude[500];
    L1BASEGMIXCAL_S3_SCANSTATUS scanStatus;
    L1BASEGMIXCAL_S3_CALIBRATION calibration;
    float incidenceAngle[500];
    float Tb[500][9];
} L1BASEGMIXCAL_S3;
#endif

#ifndef _L1BASEGMIXCAL_S2_SUNDATA_
define _L1BASEGMIXCAL_S2_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1BASEGMIXCAL_S2_SUNDATA;
#endif

#ifndef _L1BASEGMIXCAL_S2_CALCOUNTS_
define _L1BASEGMIXCAL_S2_CALCOUNTS_

typedef struct {
    unsigned short hotLoadReading[4][65];
    unsigned short coldLoadReading[4][85];
    unsigned short hotLoadnDiodeReading[4][65];
    unsigned short coldLoadnDiodeReading[4][85];
    unsigned short hotLoadThermisterCount[11];
} L1BASEGMIXCAL_S2_CALCOUNTS;
#endif
#ifndef _L1BASEGMIXCAL_S2_CAL2_
define _L1BASEGMIXCAL_S2_CAL2_

typedef struct {
    unsigned short trayTemperatureCount;
    float trayTemperature;
    unsigned short moonIndex[4];
    float noiseDiodeTemp[6];
    unsigned short TEMP_CALRES_2;
    unsigned short TEMP_CALRES_1;
    unsigned short RS_CALRES_2;
    unsigned short RS_CALRES_1;
    unsigned short BATC_CALRES_2;
    unsigned short BATC_CALRES_1;
    unsigned short NDIODE_MODE;
    unsigned short RSST_NDIODE_ST;
    unsigned short NDIODE10GHZNUM;
    float hotLoadThermisterTemp[4][11];
    unsigned short TEMP_CALRES_4;
    unsigned short TEMP_CALRES_5;
    unsigned short TEMP_CALRES_6;
    float TEMP_89GHZ_L0;
    float TEMP_166GHZ_L0;
    float TEMP_183GHZ_L0;
    float TEMP_V89GHZMXR;
    float TEMP_H89GHZMXR;
    float TEMP_V166GHZMXR;
    float TEMP_H166GHZMXR;
    float TEMP_183GHZMXR;
    float TEMP_RS_MR1;
    float TEMP_RS_MR2;
    float MR_ICA_TEMP;
    float MR_LR_LEFT_TEMP;
    float MR_LR_RGHT_TEMP;
    float MR_LR_LOWR_TEMP;
    float CSR_TEMP1;
    float CSR_TEMP2;
    float onOrbitDiodeExcessTemp[4];
    unsigned short WarmIntrusionToColdViewIndex[85][4];
} L1BASEGMIXCAL_S2_CAL2;

#endif
#ifndef _L1BASEGMIXCAL_S2_CALIBRATION_
#define _L1BASEGMIXCAL_S2_CALIBRATION_

typedef struct {
    float hotLoadTemp[4];
    float coldSkyTemp[4];
    float onOrbitNonLinearity[4];
    float derivedNonLinearity[4];
    float meanHotLoadCount[4];
    float meanHotLoadCntnDiode[4];
    float meanColdSkyCount[4];
    float meanColdSkyCntnDiode[4];
    float diodeExcessTemp[4];
    float gain[4][2];
    float offset[4][2];
    float nonLinearGain[4];
    short calibrationQCflag;
    short diodeFlag;
    float receiverTemp[4];
    float receiverGain[4];
} L1BASEGMIXCAL_S2_CALIBRATION;

#endif

#ifdef _L1BASEGMIXCAL_S2_NAV2_
#define _L1BASEGMIXCAL_S2_NAV2_

typedef struct {
    unsigned short SCE_SELECTION;
    unsigned short SCE_RATE;
} L1BASEGMIXCAL_S2_NAV2;

#endif

#ifdef _L1BASEGMIXCAL_S2_NEDTINFO_
#define _L1BASEGMIXCAL_S2_NEDTINFO_

typedef struct {
    float NEDTinfo[4];
} L1BASEGMIXCAL_S2_NEDTINFO;

#endif
#ifndef _L1BASEGMIXCAL_S2_SAMPLEHEADER_
#define _L1BASEGMIXCAL_S2_SAMPLEHEADER_

typedef struct {
    signed char blanking;
    short earthViewFirstSample;
    short sampleNumber[4][4];
    unsigned int tachSeconds[32];
    unsigned short tachSubSeconds[32];
    unsigned int indexPulseSeconds;
    unsigned short indexPulseSubSeconds;
} L1BASEGMIXCAL_S2_SAMPLEHEADER;
#endif

#endif

#ifndef _L1BASEGMIXCAL_S2_SCANSTATUS_
#define _L1BASEGMIXCAL_S2_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMIXCAL_S2_SCANSTATUS;
#endif

#ifndef _L1BASEGMIXCAL_S2_
#define _L1BASEGMIXCAL_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    L1BASEGMIXCAL_S2_SCANSTATUS scanStatus;
} L1BASEGMIXCAL_S2;
L1BASEGMIXCAL_S2_SAMPLEHEADER sampleHeader;
L1BASEGMIXCAL_S2_NEDTINFO NEDTinfo;
NAVIGATION navigation;
L1BASEGMIXCAL_S2_NAV2 nav2;
L1BASEGMIXCAL_S2_CALIBRATION calibration;
L1BASEGMIXCAL_S2_CAL2 cal2;
float moonVectorInstFrame[3];
L1BASEGMIXCAL_S2_CALCOUNTS calCounts;
L1BASEGMIXCAL_S2_SUNDATA sunData;
float incidenceAngle[221];
float satAzimuthAngle[221];
float solarZenAngle[221];
float solarAzimuthAngle[221];
float sunGlintAngle[221];
float magneticFieldVector[3];
float TAMmagneticFieldVector[3];
unsigned short earthViewCounts[221][4];
float Tb[221][4];
short RFIFlag[221][2];
} L1BASEGMIXCAL_S2;

#endif

#ifndef _L1BASEGMIXCAL_S1_SUNDATA_
#define _L1BASEGMIXCAL_S1_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1BASEGMIXCAL_S1_SUNDATA;

#endif

#ifndef _L1BASEGMIXCAL_S1_CALCOUNTS_
#define _L1BASEGMIXCAL_S1_CALCOUNTS_

typedef struct {

}}
unsigned short hotLoadReading[9][65];
unsigned short coldLoadReading[9][85];
unsigned short hotLoadnDiodeReading[9][65];
unsigned short coldLoadnDiodeReading[9][85];
unsigned short hotLoadThermisterCount[11];
} L1BASEGMIXCAL_S1_CALCOUNTS;

#endif

#ifndef _L1BASEGMIXCAL_S1_CAL2_
define _L1BASEGMIXCAL_S1_CAL2_

typedef struct {
    unsigned short trayTemperatureCount;
    float trayTemperature;
    unsigned short moonIndex[9];
    float noiseDiodeTemp[6];
    unsigned short TEMP_CALRES_2;
    unsigned short TEMP_CALRES_1;
    unsigned short RS_CALRES_2;
    unsigned short RS_CALRES_1;
    unsigned short BATC_CALRES_2;
    unsigned short BATC_CALRES_1;
    unsigned short NDIODE_MODE;
    unsigned short RSST_NDIODE_ST;
    unsigned short NDIODE10GHZNUM;
    float hotLoadThermisterTemp[9][11];
    unsigned short TEMP_CALRES_4;
    unsigned short TEMP_CALRES_5;
    unsigned short TEMP_CALRES_6;
    float TEMP_89GHZ_LO;
    float TEMP_166GHZ_LO;
    float TEMP_183GHZ_LO;
    float TEMP_V89GHZMXR;
    float TEMP_H89GHZMXR;
    float TEMP_V166GHZMXR;
    float TEMP_H166GHZMXR;
    float TEMP_183GHZMXR;
    float TEMP_RS_MR1;
    float TEMP_RS_MR2;
    float MR_ICA_TEMP;
    float MR_LR_LEFT_TEMP;
    float MR_LR_RGHT_TEMP;
}
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

float MR_LR_LOWR_TEMP;
float CSR_TEMP1;
float CSR_TEMP2;
float onOrbitDiodeExcessTemp[9];
unsigned short WarmIntrusionToColdViewIndex[85][9];
} L1BASEGMIXCAL_S1_CAL2;

#endif

#ifndef _L1BASEGMIXCAL_S1_CALIBRATION_
#define _L1BASEGMIXCAL_S1_CALIBRATION_

typedef struct {
    float hotLoadTemp[9];
    float coldSkyTemp[9];
    float onOrbitNonLinearity[9];
    float derivedNonLinearity[9];
    float meanHotLoadCount[9];
    float meanHotLoadCntnDiode[9];
    float meanColdSkyCount[9];
    float meanColdSkyCntnDiode[9];
    float diodeExcessTemp[9];
    float gain[9][2];
    float offset[9][2];
    float nonLinearGain[9];
    short calibrationQCflag;
    short diodeFlag;
    float receiverTemp[9];
    float receiverGain[9];
} L1BASEGMIXCAL_S1_CALIBRATION;

#endif

#ifndef _L1BASEGMIXCAL_S1_NAV2_
#define _L1BASEGMIXCAL_S1_NAV2_

typedef struct {
    unsigned short SCE_SELECTION;
    unsigned short SCE_RATE;
} L1BASEGMIXCAL_S1_NAV2;

#endif
```c
#ifndef _NAVIGATION_
define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;

#endif

#ifndef _L1BASEGMIXCAL_S1_NEDTINFO_
define _L1BASEGMIXCAL_S1_NEDTINFO_

typedef struct {
    float NEDTinfo[9];
} L1BASEGMIXCAL_S1_NEDTINFO;

#endif

#ifndef _L1BASEGMIXCAL_S1_SAMPLEHEADER_
define _L1BASEGMIXCAL_S1_SAMPLEHEADER_

typedef struct {
    signed char blanking;
    short earthViewFirstSample;
    short sampleNumber[9][4];
    unsigned int tachSeconds[32];
    unsigned short tachSubSeconds[32];
    unsigned int indexPulseSeconds;
    unsigned short indexPulseSubSeconds;
} L1BASEGMIXCAL_S1_SAMPLEHEADER;

#endif
```
typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BASEGMIXCAL_S1_SCANSTATUS;

#endif

#endif

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#endif
typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    L1BASEGMIXCAL_S1_SCANSTATUS scanStatus;
    L1BASEGMIXCAL_S1_SAMPLEHEADER sampleHeader;
    L1BASEGMIXCAL_S1_NEDTINFO NEDTinfo;
    NAVIGATION navigation;
    L1BASEGMIXCAL_S1_NAV2 nav2;
    L1BASEGMIXCAL_S1_CALIBRATION calibration;
    L1BASEGMIXCAL_S1_CAL2 cal2;
    float moonVectorInstFrame[3];
    L1BASEGMIXCAL_S1_CALCOUNTS calCounts;
    L1BASEGMIXCAL_S1_SUNDATA sunData;
    float incidenceAngle[221];
    float satAzimuthAngle[221];
    float solarZenAngle[221];
    float solarAzimuthAngle[221];
    float sunGlintAngle[221];
    float magneticFieldVector[3];
    float TAMmagneticFieldVector[3];
    unsigned short earthViewCounts[221][9];
    float Tb[221][9];
    short RFIFlag[221][5];
} L1BASEGMIXCAL_S1;

#endif

#ifndef _L1BASEGMIXCAL_SWATHS_
define _L1BASEGMIXCAL_SWATHS_

typedef struct {
    L1BASEGMIXCAL_S1 S1;
    L1BASEGMIXCAL_S2 S2;
    L1BASEGMIXCAL_S3 S3;
    L1BASEGMIXCAL_S4 S4;
} L1BASEGMIXCAL_SWATHS;

#undef _L1BASEGMIXCAL_SWATHS_
#endif

Fortran Structure Header file:
STRUCTURE /L1BASEGMIXCAL_S4_CALIBRATION/
    REAL*4 hotLoadTemp(4)
    REAL*4 coldSkyTemp(4)
    REAL*4 onOrbitNonLinearity(4)
    REAL*4 derivedNonLinearity(4)
    REAL*4 meanHotLoadCount(4)
    REAL*4 meanHotLoadCntnDiode(4)
    REAL*4 meanColdSkyCount(4)
    REAL*4 meanColdSkyCntnDiode(4)
    REAL*4 diodeExcessTemp(4)
    REAL*4 gain(2,4)
    REAL*4 offset(2,4)
    REAL*4 nonLinearGain(4)
    INTEGER*2 calibrationQCflag
    INTEGER*2 diodeFlag
    REAL*4 receiverTemp(4)
    REAL*4 receiverGain(4)
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S4_SCANSTATUS/
    BYTE dataQuality
    BYTE missing
    BYTE modeStatus
    INTEGER*2 geoError
    INTEGER*2 geoWarning
    INTEGER*2 SCorientation
    INTEGER*2 pointingStatus
    BYTE acsModeMidScan
    BYTE targetSelectionMidScan
    BYTE operationalMode
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S4/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(500)
    REAL*4 Longitude(500)
    RECORD /L1BASEGMIXCAL_S4_SCANSTATUS/ scanStatus
    RECORD /L1BASEGMIXCAL_S4_CALIBRATION/ calibration
    REAL*4 incidenceAngle(500)
    REAL*4 Tb(4,500)
END STRUCTURE
STRUCTURE /L1BASEGMI XCAL_S3_CALIBRATION/
    REAL*4 hotLoadTemp(9)
    REAL*4 coldSkyTemp(9)
    REAL*4 onOrbitNonLinearity(9)
    REAL*4 derivedNonLinearity(9)
    REAL*4 meanHotLoadCount(9)
    REAL*4 meanHotLoadCntnDiode(9)
    REAL*4 meanColdSkyCount(9)
    REAL*4 meanColdSkyCntnDiode(9)
    REAL*4 diodeExcessTemp(9)
    REAL*4 gain(2,9)
    REAL*4 offset(2,9)
    REAL*4 nonLinearGain(9)
    INTEGER*2 calibrationQCflag
    INTEGER*2 diodeFlag
    REAL*4 receiverTemp(9)
    REAL*4 receiverGain(9)
END STRUCTURE

STRUCTURE /L1BASEGMI XCAL_S3_SCANSTATUS/
    BYTE dataQuality
    BYTE missing
    BYTE modeStatus
    INTEGER*2 geoError
    INTEGER*2 geoWarning
    INTEGER*2 SCorientation
    INTEGER*2 pointingStatus
    BYTE acsModeMidScan
    BYTE targetSelectionMidScan
    BYTE operationalMode
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1BASEGMI XCAL_S3/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(500)
    REAL*4 Longitude(500)
    RECORD /L1BASEGMI XCAL_S3_SCANSTATUS/ scanStatus
    RECORD /L1BASEGMI XCAL_S3_CALIBRATION/ calibration
    REAL*4 incidenceAngle(500)
    REAL*4 Tb(9,500)
END STRUCTURE
5.7 1BASEGMIXCAL - GMI Antenna Temperatures

```plaintext
STRUCTURE /L1BASEGMIXCAL_S2_SUNDATA/
   REAL*4 solarBetaAngle
   REAL*4 phaseFromOrbitMidnight
   REAL*4 sunEarthSeparation
   REAL*4 earthAngularRadius
   REAL*4 phaseOfEclipseExit
   REAL*4 orbitRate
   REAL*4 timeSinceEclipseEntry
   REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S2_CALCOUNTS/
   INTEGER*2 hotLoadReading(65,4)
   INTEGER*2 coldLoadReading(85,4)
   INTEGER*2 hotLoadnDiodeReading(65,4)
   INTEGER*2 coldLoadnDiodeReading(85,4)
   INTEGER*2 hotLoadThermisterCount(11)
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S2_CAL2/
   INTEGER*2 trayTemperatureCount
   REAL*4 trayTemperature
   INTEGER*2 moonIndex(4)
   REAL*4 noiseDiodeTemp(6)
   INTEGER*2 TEMP_CALRES_2
   INTEGER*2 TEMP_CALRES_1
   INTEGER*2 RS_CALRES_2
   INTEGER*2 RS_CALRES_1
   INTEGER*2 BATC_CALRES_2
   INTEGER*2 BATC_CALRES_1
   INTEGER*2 NDIODE_MODE
   INTEGER*2 RSST_NDIODE_ST
   INTEGER*2 NDIODE10GHZNUM
   REAL*4 hotLoadThermisterTemp(11,4)
   INTEGER*2 TEMP_CALRES_4
   INTEGER*2 TEMP_CALRES_5
   INTEGER*2 TEMP_CALRES_6
   REAL*4 TEMP_89GHZ_LO
   REAL*4 TEMP_166GHZ_LO
   REAL*4 TEMP_183GHZ_LO
   REAL*4 TEMP_V89GHZMXR
   REAL*4 TEMP_H89GHZMXR
```
REAL*4 TEMP_V166GHZMXR
REAL*4 TEMP_H166GHZMXR
REAL*4 TEMP_183GHZMXR
REAL*4 TEMP_RS_MR1
REAL*4 TEMP_RS_MR2
REAL*4 MR_ICA_TEMP
REAL*4 MR_LR_LEFT_TEMP
REAL*4 MR_LR_RIGHT_TEMP
REAL*4 MR_LR_LOW_TEMP
REAL*4 CSR_TEMP1
REAL*4 CSR_TEMP2
REAL*4 onOrbitDiodeExcessTemp(4)
INTEGER*2 WarmIntrusionToColdViewIndex(4,85)
END STRUCTURE

STRUCTURE /L1BASEGMI0XCAL_S2_CALIBRATION/
REAL*4 hotLoadTemp(4)
REAL*4 coldSkyTemp(4)
REAL*4 onOrbitNonLinearity(4)
REAL*4 derivedNonLinearity(4)
REAL*4 meanHotLoadCount(4)
REAL*4 meanHotLoadCntnDiode(4)
REAL*4 meanColdSkyCount(4)
REAL*4 meanColdSkyCntnDiode(4)
REAL*4 diodeExcessTemp(4)
REAL*4 gain(2,4)
REAL*4 offset(2,4)
REAL*4 nonLinearGain(4)
INTEGER*2 calibrationQCflag
INTEGER*2 diodeFlag
REAL*4 receiverTemp(4)
REAL*4 receiverGain(4)
END STRUCTURE

STRUCTURE /L1BASEGMI0XCAL_S2_NAV2/
INTEGER*2 SCE_SELECTION
INTEGER*2 SCE_RATE
END STRUCTURE

STRUCTURE /L1BASEGMI0XCAL_S2_NEDTINFO/
REAL*4 NEDTinfo(4)
END STRUCTURE
STRUCTURE /L1BASEGMIXCAL_S2_SAMPLEHEADER/
  BYTE blanking
  INTEGER*2 earthViewFirstSample
  INTEGER*2 sampleNumber(4,4)
  INTEGER*4 tachSeconds(32)
  INTEGER*2 tachSubSeconds(32)
  INTEGER*4 indexPulseSeconds
  INTEGER*2 indexPulseSubSeconds
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S2_SCANSTATUS/
  BYTE dataQuality
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S2/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(221)
  REAL*4 Longitude(221)
  RECORD /L1BASEGMIXCAL_S2_SCANSTATUS/ scanStatus
  RECORD /L1BASEGMIXCAL_S2_SAMPLEHEADER/ sampleHeader
  RECORD /L1BASEGMIXCAL_S2_NEDTINFO/ NEDTinfo
  RECORD /NAVIGATION/ navigation
  RECORD /L1BASEGMIXCAL_S2_NAV2/ nav2
  RECORD /L1BASEGMIXCAL_S2_CALIBRATION/ calibration
  RECORD /L1BASEGMIXCAL_S2_CAL2/ cal2
  REAL*4 moonVectorInstFrame(3)
  RECORD /L1BASEGMIXCAL_S2_CALCOUNTS/ calCounts
  RECORD /L1BASEGMIXCAL_S2_SUNDATA/ sunData
  REAL*4 incidenceAngle(221)
  REAL*4 satAzimuthAngle(221)
  REAL*4 solarZenAngle(221)
  REAL*4 solarAzimuthAngle(221)
  REAL*4 sunGlintAngle(221)
REAL*4 magneticFieldVector(3)
REAL*4 TAMmagneticFieldVector(3)
INTEGER*2 earthViewCounts(4,221)
REAL*4 Tb(4,221)
INTEGER*2 RFIFlag(2,221)
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S1_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
  REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S1_CALCOUNTS/
  INTEGER*2 hotLoadReading(65,9)
  INTEGER*2 coldLoadReading(85,9)
  INTEGER*2 hotLoadnDiodeReading(65,9)
  INTEGER*2 coldLoadnDiodeReading(85,9)
  INTEGER*2 hotLoadThermisterCount(11)
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S1_CAL2/
  INTEGER*2 trayTemperatureCount
  REAL*4 trayTemperature
  INTEGER*2 moonIndex(9)
  REAL*4 noiseDiodeTemp(6)
  INTEGER*2 TEMP_CALRES_2
  INTEGER*2 TEMP_CALRES_1
  INTEGER*2 RS_CALRES_2
  INTEGER*2 RS_CALRES_1
  INTEGER*2 BATC_CALRES_2
  INTEGER*2 BATC_CALRES_1
  INTEGER*2 NDIODE_MODE
  INTEGER*2 RSST_NDIODE_ST
  INTEGER*2 NDIODE10GHZNUM
  REAL*4 hotLoadThermisterTemp(11,9)
  INTEGER*2 TEMP_CALRES_4
  INTEGER*2 TEMP_CALRES_5
INTEGER*2 TEMP_CALRES_6
REAL*4 TEMP_89GHZ_LO
REAL*4 TEMP_166GHZ_LO
REAL*4 TEMP_183GHZ_LO
REAL*4 TEMP_V89GHZMXR
REAL*4 TEMP_H89GHZMXR
REAL*4 TEMP_V166GHZMXR
REAL*4 TEMP_H166GHZMXR
REAL*4 TEMP_183GHZMXR
REAL*4 TEMP_RS_MR1
REAL*4 TEMP_RS_MR2
REAL*4 MR_ICA_TEMP
REAL*4 MR_LR_LEFT_TEMP
REAL*4 MR_LR_RGHT_TEMP
REAL*4 MR_LR_LOWR_TEMP
REAL*4 CSR_TEMP1
REAL*4 CSR_TEMP2
REAL*4 onOrbitDiodeExcessTemp(9)
INTEGER*2 WarmIntrusionToColdViewIndex(9, 85)
END STRUCTURE

STRUCTURE /L1BASEGMEMIXCAL_S1_CALIBRATION/
REAL*4 hotLoadTemp(9)
REAL*4 coldSkyTemp(9)
REAL*4 onOrbitNonLinearity(9)
REAL*4 derivedNonLinearity(9)
REAL*4 meanHotLoadCount(9)
REAL*4 meanHotLoadCntnDiode(9)
REAL*4 meanColdSkyCount(9)
REAL*4 meanColdSkyCntnDiode(9)
REAL*4 diodeExcessTemp(9)
REAL*4 gain(2, 9)
REAL*4 offset(2, 9)
REAL*4 nonLinearGain(9)
INTEGER*2 calibrationQCflag
INTEGER*2 diodeFlag
REAL*4 receiverTemp(9)
REAL*4 receiverGain(9)
END STRUCTURE

STRUCTURE /L1BASEGMEMIXCAL_S1_NAV2/
INTEGER*2 SCE_SELECTION
INTEGER*2 SCE_RATE
END STRUCTURE

STRUCTURE /NAVIGATION/
    REAL*4 scPos(3)
    REAL*4 scVel(3)
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 dprAlt
    REAL*4 scAttRollGeoc
    REAL*4 scAttPitchGeoc
    REAL*4 scAttYawGeoc
    REAL*4 scAttRollGeod
    REAL*4 scAttPitchGeod
    REAL*4 scAttYawGeod
    REAL*4 greenHourAng
    REAL*8 timeMidScan
    REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S1_NEDTINFO/
    REAL*4 NEDTinfo(9)
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S1_SAMPLEHEADER/
    BYTE blanking
    INTEGER*2 earthViewFirstSample
    INTEGER*2 sampleNumber(4,9)
    INTEGER*4 tachSeconds(32)
    INTEGER*2 tachSubSeconds(32)
    INTEGER*4 indexPulseSeconds
    INTEGER*2 indexPulseSubSeconds
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S1_SCANSTATUS/
    BYTE dataQuality
    BYTE missing
    BYTE modeStatus
    INTEGER*2 geoError
    INTEGER*2 geoWarning
    INTEGER*2 SCorientation
    INTEGER*2 pointingStatus
    BYTE acsModeMidScan

BYTE targetSelectionMidScan
BYTE operationalMode
REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
   INTEGER*2 Year
   BYTE Month
   BYTE DayOfMonth
   BYTE Hour
   BYTE Minute
   BYTE Second
   INTEGER*2 MilliSecond
   INTEGER*2 DayOfYear
   REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASEGMIXCAL_S1/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(221)
   REAL*4 Longitude(221)
   RECORD /L1BASEGMIXCAL_S1_SCANSTATUS/ scanStatus
   RECORD /L1BASEGMIXCAL_S1_SAMPLEHEADER/ sampleHeader
   RECORD /L1BASEGMIXCAL_S1_NEDTINFO/ NEDTinfo
   RECORD /NAVIGATION/ navigation
   RECORD /L1BASEGMIXCAL_S1_NAV2/ nav2
   RECORD /L1BASEGMIXCAL_S1_CALIBRATION/ calibration
   RECORD /L1BASEGMIXCAL_S1_CAL2/ cal2
   REAL*4 moonVectorInstFrame(3)
   RECORD /L1BASEGMIXCAL_S1_CALCOUNTS/ calCounts
   RECORD /L1BASEGMIXCAL_S1_SUNDATA/ sunData
   REAL*4 incidenceAngle(221)
   REAL*4 satAzimuthAngle(221)
   REAL*4 solarZenAngle(221)
   REAL*4 solarAzimuthAngle(221)
   REAL*4 sunGlintAngle(221)
   REAL*4 magneticFieldVector(3)
   REAL*4 TAMmagneticFieldVector(3)
   INTEGER*2 earthViewCounts(9,221)
   REAL*4 Tb(9,221)
   INTEGER*2 RFIFlag(5,221)
END STRUCTURE
STRUCTURE /L1BASEGMIXCAL_SWATHS/
   RECORD /L1BASEGMIXCAL_S1/ S1;
   RECORD /L1BASEGMIXCAL_S2/ S2;
   RECORD /L1BASEGMIXCAL_S3/ S3;
   RECORD /L1BASEGMIXCAL_S4/ S4;
END STRUCTURE

5.8 1BASESSMI - SSMI base

1BASESSMI contains both antenna temperature and brightness temperature. These files contain all the information from the TDR files plus additions Wesley Berg made creating fcdr files plus the isMissing flag. 1BASESSMI is written as one swath, but the 1C product will have 2 swaths.

NOTE: the filespec does not draw the swath. NOTE: ranges may be wrong

Dimension definitions:

- nscan var Number of scans in the granule.
- npixel1 64 Number pixels for low res.
- npixel2 128 Number pixels for high res.
- nchan1 5 Number pixels for low res.
- nchan2 2 Number pixels for high res.
- scandim 2 Number scan array dim (A scan, B scan).
- loaddim 5 Number of (hot or cold) load array dimensions.
- timedim 6 Number of time array dim (year, month, day, hour, minute, second, and millisecond)

Figure 219 through Figure 226 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.
Figure 219: Data Format Structure for 1BASESSMI, SSMI base
continued from last figure

- scロン_or&_ 4 bytes Array: nscan
- sc Lor_ 4 bytes Array: nscan
- sc time_ 8 bytes Array: nscan
- scpacecraft_l&_ 4 bytes Array: scdim x nscan
- scpacecraft lon_ 4 bytes Array: scdim x nscan
- scspacecraft alt_ 4 bytes Array: scdim x nscan
- sc solar beta angle_ 4 bytes Array: scdim x nscan
- scsun elevation angle_ 4 bytes Array: scdim x nscan
- scsun azimuth angle_ 4 bytes Array: scdim x nscan
- scphase orbit midnight_ 4 bytes Array: scdim x nscan
- scsun earth separation_ 4 bytes Array: scdim x nscan
- scp& earth radius_ 4 bytes Array: scdim x nscan
- scphase eclipse exit_ 4 bytes Array: scdim x nscan
- sceclipse status_ 1 byte Array: scdim x nscan
- sctime since eclipse_ 4 bytes Array: scdim x nscan
- scautogain1a_ 4 bytes Array: nscan
- scautogain2a_ 4 bytes Array: nscan
- scautogain3a_ 4 bytes Array: nscan
- scautogain1b_ 4 bytes Array: nscan
- scautogain2b_ 4 bytes Array: nscan
- scautogain3b_ 4 bytes Array: nscan
- scrfmร & temp_ 4 bytes Array: nscan
- sctrad temp_ 4 bytes Array: nscan

continued on next figure

Figure 220: Data Format Structure for 1BASESSMI, SSMI base
5.8 1BASESSMI - SSMI base

continued from last figure

Figure 221: Data Format Structure for 1BASESSMI, SSMI base
continued from last figure

Figure 222: Data Format Structure for 1BASESSMI, SSMI base
### 5.8 1BASESSMI - SSMI base

**Figure 223: Data Format Structure for 1BASESSMI, SSMI base**

- **quality_hires**: 1 byte, Array: npixel2 x scandim x nscan
- **sun_glint_hires**: 4 bytes, Array: npixel2 x scandim x nscan
- **ea_hires**: 4 bytes, Array: npixel2 x scandim x nscan
- **sunzenith_hires**: 4 bytes, Array: npixel2 x scandim x nscan
- **scazimuth_hires**: 4 bytes, Array: npixel2 x scandim x nscan
- **sunazimuth_hires**: 4 bytes, Array: npixel2 x scandim x nscan
- **ta85v**: 4 bytes, Array: npixel2 x scandim x nscan
- **tb85v**: 4 bytes, Array: npixel2 x scandim x nscan
- **ta85h**: 4 bytes, Array: npixel2 x scandim x nscan
- **tb85h**: 4 bytes, Array: npixel2 x scandim x nscan
- **cold_load_19v**: 4 bytes, Array: loaddim x nscan
- **hot_load_19v**: 4 bytes, Array: loaddim x nscan
- **cold_load_19h**: 4 bytes, Array: loaddim x nscan
- **hot_load_19h**: 4 bytes, Array: loaddim x nscan
- **cold_load_22v**: 4 bytes, Array: loaddim x nscan
- **hot_load_22v**: 4 bytes, Array: loaddim x nscan
- **cold_load_37v**: 4 bytes, Array: loaddim x nscan
- **hot_load_37v**: 4 bytes, Array: loaddim x nscan
- **cold_load_37h**: 4 bytes, Array: loaddim x nscan
- **hot_load_37h**: 4 bytes, Array: loaddim x nscan
- **cold_load_85va**: 4 bytes, Array: loaddim x nscan
- **hot_load_85va**: 4 bytes, Array: loaddim x nscan
- **cold_load_85ha**: 4 bytes, Array: loaddim x nscan

continued on next figure

...
continued from last figure

Figure 224: Data Format Structure for 1BASESSMI, SSMI base

Figure 225: Data Format Structure for 1BASESSMI, baseHeader
5.8 1BASESSMI - SSMI base

**Figure 226: Data Format Structure for 1BASESSMI, ScanTime**

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**baseHeader** (Group)

- **spacecraft_id** (4-byte integer, array size: 1):
  Satellite ID number.
- **ascend_time** (4-byte integer, array size: 6):
  Ascending time.
- **TLE_xtime** (8-byte float, array size: 1):
  TLE time.
- **TLE_date** (4-byte integer, array size: 6):
  TLE date time arrays.
- **TLE_time** (8-byte float, array size: 1):
  TLE time as in two line element.
- **quality_tests** (4-byte integer, array size: 9):
  Results from Wes Berg’s fcdr quality control tests.
- **nominal_elevation_angle** (4-byte float, array size: 1):
  Nominal sensor elevation angle.
- **delta_elevation_angle** (4-byte float, array size: 1):
  Offset in the sensor elevation angle from nominal.
- **spacecraft_roll** (4-byte float, array size: 1):
  Spacecraft roll angle offset from nominal.
spacecraft_pitch (4-byte float, array size: 1):
Spacecraft pitch angle offset from nominal.

spacecraft_yaw (4-byte float, array size: 1):
Spacecraft yaw angle offset from nominal.

S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
   -9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
   -99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
   -99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
   -99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
   -99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
   -99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
   -9999 Missing value
DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

isMissing (1-byte integer, array size: nscan):
Missing scan flag.

scan_time (8-byte float, array size: scandim x nscan):
A and B scan start time in seconds since 1987-01-01T00:00:00.00Z.

orbit (8-byte float, array size: scandim x nscan):
Fractional orbit number for A and B scans.

scposx_gci1 (8-byte float, array size: scandim x nscan):
Orbital Position Vector X in Geocentric Inertial Coordinates for the first pixel.

scposy_gci1 (8-byte float, array size: scandim x nscan):
Orbital Position Vector Y in Geocentric Inertial Coordinates for the first pixel.

scposz_gci1 (8-byte float, array size: scandim x nscan):
Orbital Position Vector Z in Geocentric Inertial Coordinates for the first pixel.

scvelx_gci1 (8-byte float, array size: scandim x nscan):
Orbital Velocity Vector X in Geocentric Inertial Coordinates for the first pixel.

scvely_gci1 (8-byte float, array size: scandim x nscan):
Orbital Velocity Vector Y in Geocentric Inertial Coordinates for the first pixel.

scvelz_gci1 (8-byte float, array size: scandim x nscan):
Orbital Velocity Vector Z in Geocentric Inertial Coordinates for the first pixel.

scposx_gci2 (8-byte float, array size: scandim x nscan):
Orbital Position Vector X in Geocentric Inertial Coordinates for the last pixel.

scposy_gci2 (8-byte float, array size: scandim x nscan):
Orbital Position Vector Y in Geocentric Inertial Coordinates for the last pixel.

scposz_gci2 (8-byte float, array size: scandim x nscan):
Orbital Position Vector Z in Geocentric Inertial Coordinates for the last pixel.

scvelx_gci2 (8-byte float, array size: scandim x nscan):
Orbital Velocity Vector X in Geocentric Inertial Coordinates for the last pixel.

scvely_gci2 (8-byte float, array size: scandim x nscan):
Orbital Velocity Vector Y in Geocentric Inertial Coordinates for the last pixel.

scvelz_gci2 (8-byte float, array size: scandim x nscan):
Orbital Velocity Vector Z in Geocentric Inertial Coordinates for the last pixel.
**sclat_orig** (4-byte float, array size: nscan):
Original spacecraft latitude from TDR data.

**sclon_orig** (4-byte float, array size: nscan):
Original spacecraft longitude from TDR data.

**scalt_orig** (4-byte float, array size: nscan):
Original spacecraft altitude from TDR data.

**sctime_orig** (8-byte float, array size: nscan):
Original spacecraft time from TDR data.

**spacecraft_lat** (4-byte float, array size: scandim x nscan):
Computed spacecraft latitude using TLE corresponding to A and B scan time.

**spacecraft_lon** (4-byte float, array size: scandim x nscan):
Computed spacecraft longitude using TLE corresponding to A and B scan time.

**spacecraft_alt** (4-byte float, array size: scandim x nscan):
Computed Spacecraft altitude using TLE corresponding to A and B scan time.

**solar_beta_angle** (4-byte float, array size: scandim x nscan):
Solar beta angle for A and B scans.

**sun_elevation_angle** (4-byte float, array size: scandim x nscan):
Sun elevation angle from spacecraft for A and B scans.

**sun_azimuth_angle** (4-byte float, array size: scandim x nscan):
Sun azimuth angle from spacecraft for A and B scans.

**phase_orbit_midnight** (4-byte float, array size: scandim x nscan):
Phase from orbit midnight for A and B scans.

**sun_earth_separation** (4-byte float, array size: scandim x nscan):
Sun Earth separation angle for A and B scans.

**earth_angle_radius** (4-byte float, array size: scandim x nscan):
Earth angle radius for A and B scans.

**phase_eclipse_exit** (4-byte float, array size: scandim x nscan):
Orbit phase for eclipse exit or entr for A and B scans.

**eclipse_status** (1-byte integer, array size: scandim x nscan):
Eclipse status (0=sun, 1=shadow) for A and B scans.

**time_since_eclipse** (4-byte float, array size: scandim x nscan):
Time since eclipse for A and B scans.

**autogain1a** (4-byte float, array size: nscan):
Auto gain control setting 1 (scan A) from scan header 1.

**autogain2a** (4-byte float, array size: nscan):
Auto gain control setting 2 (scan A) from scan header 1.

**autogain3a** (4-byte float, array size: nscan):
Auto gain control setting 3 (scan A) from scan header 1.
autogain1b (4-byte float, array size: nscan):
Auto gain control setting 1 (scan B) from scan header 1.

autogain2b (4-byte float, array size: nscan):
Auto gain control setting 2 (scan B) from scan header 1.

autogain3b (4-byte float, array size: nscan):
Auto gain control setting 3 (scan B) from scan header 1.

rfmxrtemp (4-byte float, array size: nscan):
RF Mixer temperature.

forwardradtemp (4-byte float, array size: nscan):
Forward Radiator Temperature.

hotload1 (4-byte float, array size: nscan):
Hot Load Thermal Temperature 1.

hotload2 (4-byte float, array size: nscan):
Hot Load Thermal Temperature 2.

hotload3 (4-byte float, array size: nscan):
Hot Load Thermal Temperature 3.

refvolt1 (4-byte float, array size: nscan):
Reference Voltage 1.

refvolt2 (4-byte float, array size: nscan):
Reference Voltage 2.

slope19v (4-byte float, array size: nscan):
19.35 GHz V-Pol channel slope.

slope19h (4-byte float, array size: nscan):
19.35 GHz H-Pol channel slope.

slope22v (4-byte float, array size: nscan):
22.235 GHz V-Pol channel slope.

slope37h (4-byte float, array size: nscan):
37.0 GHz H-Pol channel slope.

slope37v (4-byte float, array size: nscan):
37.0 GHz V-Pol channel slope.

slope85h (4-byte float, array size: nscan):
85.5 GHz H-Pol channel slope.

slope85v (4-byte float, array size: nscan):
85.5 GHz V-Pol channel slope.

offset19v (4-byte float, array size: nscan):
19.35 GHz V-Pol channel offset.

offset19h (4-byte float, array size: nscan):
19.35 GHz H-Pol channel offset.
**offset22v** (4-byte float, array size: nscan):
22.235 GHz V-Pol channel offset.

**offset37h** (4-byte float, array size: nscan):
37.0 GHz H-Pol channel offset.

**offset37v** (4-byte float, array size: nscan):
37.0 GHz V-Pol channel offset.

**offset85h** (4-byte float, array size: nscan):
85.5 GHz H-Pol channel offset.

**offset85v** (4-byte float, array size: nscan):
85.5 GHz V-Pol channel offset.

**scan_datetime** (4-byte float, array size: timedim x scandim x nscan):
Scan date time array.

**lon_lores_orig** (4-byte float, array size: npixel1 x nscan):
Original pixel longitude from TDR data for low resolution channels.

**lat_lores_orig** (4-byte float, array size: npixel1 x nscan):
Original pixel latitude from TDR data for low resolution channels.

**lon_lores** (4-byte float, array size: npixel1 x nscan):
TLE Computed longitude for low resolution channels.

**lat_lores** (4-byte float, array size: npixel1 x nscan):
TLE Computed latitude for low resolution channels.

**sfctype_lores** (4-byte integer, array size: npixel1 x nscan):
Surface type for low resolution channels.

**quality_lores** (1-byte integer, array size: npixel1 x nscan):
Quality flag for low resolution channels. 0=Good data, 1-99=Minor issue (use with caution), 100-255=Major issue (set to missing).

- 0  Good data
- 1  Possible sun glint
- 2  Climatology check warning (19V Channel)
- 3  Climatology check warning (19H Channel)
- 4  Climatology check warning (22V Channel)
- 5  Climatology check warning (37V Channel)
- 6  Climatology check warning (37H Channel)
- 7  Climatology check warning (85V Channel)
- 8  Climatology check warning (85H Channel)
- 9  Climatology check warning (Multiple low-res channels)
- 10 Climatology check warning (Multiple high-res channels)
- 11 Warning of increased noise in 85V channel on DMSP F08
- 12 RADCAL correction applied to Tb22v (do not use for climate)
Correction made to $Ta$ by correcting for spikes in warm/cold load cal data
Data is missing from file or unreadable
Geolocation check flagged in input BASE file
Climatology check flagged in input BASE file
Climatology check failed (19V Channel)
Climatology check failed (19H Channel)
Climatology check failed (22V Channel)
Climatology check failed (37V Channel)
Climatology check failed (37H Channel)
Climatology check failed (85V Channel)
Climatology check failed (85H Channel)
Climatology check failed (Multiple low-res channels)
Climatology check failed (Multiple high-res channels)
Distance between pixels is nonphysical
Antenna temperatures are lt 50 or gt 350 K
Lat/Lon values are out of range
Failure of 85V channel on DMSP F08
Failure of 85V and increased noise in 85H on DMSP F08
Failure of both 85V and 85H channels on DMSP F08
Invalid scan time
$Ta$ set to missing due to bad cal data
All data set to missing

`sun_glint_lores` (4-byte float, array size: npixel1 x nscan):
Sun glint angle for low resolution channels.

`eia_lores` (4-byte float, array size: npixel1 x nscan):
Earth incidence angle for low resolution channels.

`sunzenith_lores` (4-byte float, array size: npixel1 x nscan):
Sun zenith angle for low resolution channels.

`scazimuth_lores` (4-byte float, array size: npixel1 x nscan):
Satellite azimuth angle for low resolution channels.

`sunazimuth_lores` (4-byte float, array size: npixel1 x nscan):
Sun azimuth angle for low resolution channels.

`ta19v` (4-byte float, array size: npixel1 x nscan):
19.35 GHz V-Pol Antenna Temperature.

`tb19v` (4-byte float, array size: npixel1 x nscan):
19.35 GHz V-Pol Brightness Temperature.

`ta19h` (4-byte float, array size: npixel1 x nscan):
19.35 GHz H-Pol Antenna Temperature.

`tb19h` (4-byte float, array size: npixel1 x nscan):
19.35 GHz H-Pol Brightness Temperature.
\( \text{ta22v} \) (4-byte float, array size: npixel1 x nscan): 22.235 GHz V-Pol Antenna Temperature.

\( \text{tb22v} \) (4-byte float, array size: npixel1 x nscan): 22.235 GHz V-Pol Brightness Temperature.

\( \text{ta37v} \) (4-byte float, array size: npixel1 x nscan): 37.0 GHz V-Pol Antenna Temperature.

\( \text{tb37v} \) (4-byte float, array size: npixel1 x nscan): 37.0 GHz V-Pol Brightness Temperature.

\( \text{ta37h} \) (4-byte float, array size: npixel1 x nscan): 37.0 GHz H-Pol Antenna Temperature.

\( \text{tb37h} \) (4-byte float, array size: npixel1 x nscan): 37.0 GHz H-Pol Brightness Temperature.

\( \text{lon} \text{hires} \text{orig} \) (4-byte float, array size: npixel2 x scandim x nscan): Original pixel longitude from TDR data for high resolution channels.

\( \text{lat} \text{hires} \text{orig} \) (4-byte float, array size: npixel2 x scandim x nscan): Original pixel latitude from TDR data for high resolution channels.

\( \text{lon} \text{hires} \) (4-byte float, array size: npixel2 x scandim x nscan): TLE Computed longitude for high resolution channels.

\( \text{lat} \text{hires} \) (4-byte float, array size: npixel2 x scandim x nscan): TLE Computed latitude for high resolution channels.

\( \text{sfctype} \text{hires} \) (4-byte integer, array size: npixel2 x scandim x nscan): Surface type for high resolution channels.

\( \text{quality} \text{hires} \) (1-byte integer, array size: npixel2 x scandim x nscan): Quality flag for high resolution channels. See quality_lores description for flag definition.

\( \text{sun} \text{glint} \text{hires} \) (4-byte float, array size: npixel2 x scandim x nscan): Sun glint angle for high resolution channels.

\( \text{eia} \text{hires} \) (4-byte float, array size: npixel2 x scandim x nscan): Earth incidence angle for high resolution channels.

\( \text{sunzenith} \text{hires} \) (4-byte float, array size: npixel2 x scandim x nscan): Sun zenith angle for high resolution channels.

\( \text{scazimuth} \text{hires} \) (4-byte float, array size: npixel2 x scandim x nscan): Satellite azimuth angle for high resolution channels.

\( \text{sunazimuth} \text{hires} \) (4-byte float, array size: npixel2 x scandim x nscan): Sun azimuth angle for high resolution channels.

\( \text{ta85v} \) (4-byte float, array size: npixel2 x scandim x nscan): 85.5 GHz V-Pol Antenna Temperature.

\( \text{tb85v} \) (4-byte float, array size: npixel2 x scandim x nscan): 85.5 GHz V-Pol Brightness Temperature.
5.8 1BASESSMI - SSMI base

ta85h (4-byte float, array size: npixel2 x scandim x nscan): 85.5 GHz H-Pol Antenna Temperature.

tb85h (4-byte float, array size: npixel2 x scandim x nscan): 85.5 GHz H-Pol Brightness Temperature.

cold_load_19v (4-byte float, array size: loaddim x nscan): 19.35 GHz V-Pol Cold Load Reading.

hot_load_19v (4-byte float, array size: loaddim x nscan): 19.35 GHz V-Pol Hot Load Reading.

cold_load_19h (4-byte float, array size: loaddim x nscan): 19.35 GHz H-Pol Cold Load Reading.

hot_load_19h (4-byte float, array size: loaddim x nscan): 19.35 GHz H-Pol Hot Load Reading.

cold_load_22v (4-byte float, array size: loaddim x nscan): 22.235 GHz V-Pol Cold Load Reading.

hot_load_22v (4-byte float, array size: loaddim x nscan): 22.235 GHz V-Pol Hot Load Reading.

cold_load_37v (4-byte float, array size: loaddim x nscan): 37.0 GHz V-Pol Cold Load Reading.

hot_load_37v (4-byte float, array size: loaddim x nscan): 37.0 GHz V-Pol Hot Load Reading.

cold_load_37h (4-byte float, array size: loaddim x nscan): 37.0 GHz H-Pol Cold Load Reading.

hot_load_37h (4-byte float, array size: loaddim x nscan): 37.0 GHz H-Pol Hot Load Reading.

cold_load_85va (4-byte float, array size: loaddim x nscan): 85.5 GHz V-Pol Cold Load Reading (A-scan).

hot_load_85va (4-byte float, array size: loaddim x nscan): 85.5 GHz V-Pol Hot Load Reading (A-scan).

cold_load_85ha (4-byte float, array size: loaddim x nscan): 85.5 GHz H-Pol Cold Load Reading (A-scan).

hot_load_85ha (4-byte float, array size: loaddim x nscan): 85.5 GHz H-Pol Hot Load Reading (A-scan).

cold_load_85vb (4-byte float, array size: loaddim x nscan): 85.5 GHz V-Pol Cold Load Reading (B-scan).

hot_load_85vb (4-byte float, array size: loaddim x nscan): 85.5 GHz V-Pol Hot Load Reading (B-scan).

cold_load_85hb (4-byte float, array size: loaddim x nscan): 85.5 GHz H-Pol Cold Load Reading (B-scan).
**hot_load_85hb** (4-byte float, array size: loaddim x nscan):
85.5 GHz H-Pol Hot Load Reading (B-scan).

**C Structure Header file:**

```c
#ifndef _TK_1BASESSMI_H_
define _TK_1BASESSMI_H_
#endif

#ifndef _SCANTIME_
define _SCANTIME_
#endif

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L1BASESSMI_S1_
define _L1BASESSMI_S1_
#endif

typedef struct {
    SCANTIME ScanTime;
    signed char isMissing;
    double scan_time[2];
    double orbit[2];
    double scposx_gci1[2];
    double scposy_gci1[2];
    double scposz_gci1[2];
    double scvelx_gci1[2];
    double scvely_gci1[2];
    double scvelz_gci1[2];
    double scvelx_gci2[2];
    double scvely_gci2[2];
    double scvelz_gci2[2];
} L1BASESSMI_S1;
```
double scvelz_gci2[2];
float sclat_orig;
float sclon_orig;
float scalt_orig;
double sctime_orig;
float spacecraft_lat[2];
float spacecraft_lon[2];
float spacecraft_alt[2];
float solar_beta_angle[2];
float sun_elevation_angle[2];
float sun_azimuth_angle[2];
float phase_orbit_midnight[2];
float sun_earth_separation[2];
float earth_angle_radius[2];
float phase_eclipse_exit[2];
signed char eclipse_status[2];
float time_since_eclipse[2];
float autogain1a;
float autogain2a;
float autogain3a;
float autogain1b;
float autogain2b;
float autogain3b;
float rfmxrtemp;
float forwardradtemp;
float hotload1;
float hotload2;
float hotload3;
float refvolt1;
float refvolt2;
float slope19v;
float slope19h;
float slope22v;
float slope37h;
float slope37v;
float slope85h;
float slope85v;
float offset19v;
float offset19h;
float offset22v;
float offset37v;
float offset37h;
float offset85h;
float offset85v;
float scan_datetime[2][6];
float lon_lores_orig[64];
float lat_lores_orig[64];
float lon_lores[64];
float lat_lores[64];
int sfctype_lores[64];
signed char quality_lores[64];
float sun_glint_lores[64];
float eia_lores[64];
float sunzenith_lores[64];
float scazimuth_lores[64];
float sunazimuth_lores[64];
float ta19v[64];
float tb19v[64];
float ta19h[64];
float tb19h[64];
float ta22v[64];
float tb22v[64];
float ta37v[64];
float tb37v[64];
float ta37h[64];
float tb37h[64];
float lon_hires_orig[2][128];
float lat_hires_orig[2][128];
float lon_hires[2][128];
float lat_hires[2][128];
int sfctype_hires[2][128];
signed char quality_hires[2][128];
float sun_glint_hires[2][128];
float eia_hires[2][128];
float sunzenith_hires[2][128];
float scazimuth_hires[2][128];
float sunazimuth_hires[2][128];
float ta85v[2][128];
float tb85v[2][128];
float ta85h[2][128];
float tb85h[2][128];
float cold_load_19v[5];
float hot_load_19v[5];
float cold_load_19h[5];
float hot_load_19h[5];
float cold_load_22v[5];
float hot_load_22v[5];
float cold_load_37v[5];
float hot_load_37v[5];
float cold_load_37h[5];
float hot_load_37h[5];
float cold_load_85va[5];
float hot_load_85va[5];
float cold_load_85ha[5];
float hot_load_85ha[5];
float cold_load_85vb[5];
float hot_load_85vb[5];
float cold_load_85hb[5];
float hot_load_85hb[5];
}
} L1BASESSMI_S1;
#endif

#ifndef _L1BASESSMI_BASEHEADER_
#define _L1BASESSMI_BASEHEADER_

typedef struct {
    int spacecraft_id[1];
    int ascend_time[6];
    double TLE_xtime[1];
    int TLE_date[6];
    double TLE_time[1];
    int quality_tests[9];
    float nominal_elevation_angle[1];
    float delta_elevation_angle[1];
    float spacecraft_roll[1];
    float spacecraft_pitch[1];
    float spacecraft_yaw[1];
} L1BASESSMI_BASEHEADER;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
BYTE DayOfMonth
BYTE Hour
BYTE Minute
BYTE Second
INTEGER*2 MilliSecond
INTEGER*2 DayOfYear
REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASESSMI_S1/
RECORD /SCANTIME/ ScanTime
BYTE isMissing
REAL*8 scan_time(2)
REAL*8 orbit(2)
REAL*8 scposx_gci1(2)
REAL*8 scposy_gci1(2)
REAL*8 scposz_gci1(2)
REAL*8 scvelx_gci1(2)
REAL*8 scvely_gci1(2)
REAL*8 scvelz_gci1(2)
REAL*8 scposx_gci2(2)
REAL*8 scposy_gci2(2)
REAL*8 scposz_gci2(2)
REAL*8 scvelx_gci2(2)
REAL*8 scvely_gci2(2)
REAL*8 scvelz_gci2(2)
REAL*4 sclat_orig
REAL*4 sclon_orig
REAL*4 scalt_orig
REAL*8 sctime_orig
REAL*4 spacecraft_lat(2)
REAL*4 spacecraft_lon(2)
REAL*4 spacecraft_alt(2)
REAL*4 solar_beta_angle(2)
REAL*4 sun_elevation_angle(2)
REAL*4 sun_azimuth_angle(2)
REAL*4 phase_orbit_midnight(2)
REAL*4 sun_earth_separation(2)
REAL*4 earth_angle_radius(2)
REAL*4 phase_eclipse_exit(2)
BYTE eclipse_status(2)
REAL*4 time_since_eclipse(2)
REAL*4 autogain1a
REAL*4 autogain2a
REAL*4 autogain3a
REAL*4 autogain1b
REAL*4 autogain2b
REAL*4 autogain3b
REAL*4 rfmxrtemp
REAL*4 forwardradtemp
REAL*4 hotload1
REAL*4 hotload2
REAL*4 hotload3
REAL*4 refvolt1
REAL*4 refvolt2
REAL*4 slope19v
REAL*4 slope19h
REAL*4 slope22v
REAL*4 slope37h
REAL*4 slope37v
REAL*4 slope85h
REAL*4 slope85v
REAL*4 offset19v
REAL*4 offset19h
REAL*4 offset22v
REAL*4 offset37h
REAL*4 offset37v
REAL*4 offset85h
REAL*4 offset85v
REAL*4 scan_datetime(6,2)
REAL*4 lon_lores_orig(64)
REAL*4 lat_lores_orig(64)
REAL*4 lon_lores(64)
REAL*4 lat_lores(64)
INTEGER*4 sfctype_lores(64)
BYTE quality_lores(64)
REAL*4 sun_glint_lores(64)
REAL*4 eia_lores(64)
REAL*4 sunzenith_lores(64)
REAL*4 scazimuth_lores(64)
REAL*4 sunazimuth_lores(64)
REAL*4 ta19v(64)
REAL*4 tb19v(64)
REAL*4 ta19h(64)
REAL*4 ta19h(64)
REAL*4 ta22v(64)
REAL*4  tb22v(64)
REAL*4  ta37v(64)
REAL*4  tb37v(64)
REAL*4  ta37h(64)
REAL*4  tb37h(64)
REAL*4  lon_hires_orig(128,2)
REAL*4  lat_hires_orig(128,2)
REAL*4  lon_hires(128,2)
REAL*4  lat_hires(128,2)
INTEGER*4  sfctype_hires(128,2)
BYTE  quality_hires(128,2)
REAL*4  sun_glint_hires(128,2)
REAL*4  eia_hires(128,2)
REAL*4  sunzenith_hires(128,2)
REAL*4  sc azimuth_hires(128,2)
REAL*4  sunazimuth_hires(128,2)
REAL*4  ta85v(128,2)
REAL*4  tb85v(128,2)
REAL*4  ta85h(128,2)
REAL*4  tb85h(128,2)
REAL*4  cold_load_19v(5)
REAL*4  hot_load_19v(5)
REAL*4  cold_load_19h(5)
REAL*4  hot_load_19h(5)
REAL*4  cold_load_22v(5)
REAL*4  hot_load_22v(5)
REAL*4  cold_load_37v(5)
REAL*4  hot_load_37v(5)
REAL*4  cold_load_37h(5)
REAL*4  hot_load_37h(5)
REAL*4  cold_load_85va(5)
REAL*4  hot_load_85va(5)
REAL*4  cold_load_85ha(5)
REAL*4  hot_load_85ha(5)
REAL*4  cold_load_85vb(5)
REAL*4  hot_load_85vb(5)
REAL*4  cold_load_85vb(5)
REAL*4  hot_load_85hb(5)
REAL*4  hot_load_85hb(5)
END STRUCTURE

STRUCTURE /L1BASESSMI_BASEHEADER/
  INTEGER*4  spacecraft_id(1)
  INTEGER*4  asc end_time(6)
5.9 1BASESSMIS - SSMIS base

1BASESSMIS contains both antenna temperature and brightness temperature. These files contain all the information from the TDR files and additions Wesley Berg made creating his fcdr files plus the isMissing flag. 1BASESSMIS is written as one swath, but the 1C product will have 4 swaths.

NOTE: the filespec does not draw the swath. NOTE: ranges may be wrong

Dimension definitions:

- `nscan` var Number of scans in the granule.
- `npixel_img` 180 Number samples per imager scanline.
- `npixel_env` 90 Number samples per environmental scanline.
- `nephem` 3 Number ephem.
- `ndate` 6 Number date.
- `ntime` 7 Number of time array dim (year, month, day, hour, minute, second, and millisecond)
- `nsunvec` 3 Number sun vector.
- `nsatpos` 2 Number satellite position.
- `nchannel` 24 Number channels.
- `nwarmload` 3 Number of warm load array dimension.
- `nmuxhouse` 4 Number mux housing.
- `nbasepoint` 28 Number basepoint.
- `ntest` 8 Number test.
- `nsensor` 6 Number sensor.

Figure 227 through Figure 234 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.
Figure 227: Data Format Structure for 1BASESSMIS, SSMIS base

continued on next figure

•

•
continued from last figure

Figure 228: Data Format Structure for 1BASESSMIS, SSMIS base
continued from last figure

Figure 229: Data Format Structure for 1BASESSMIS, SSMIS base
continued from last figure

- aux_lat_kaband 4 bytes Array: nbasepoint x nscan
- aux_lon_kaband 4 bytes Array: nbasepoint x nscan
- aux_eia_kaband 4 bytes Array: nbasepoint x nscan
- aux_azimuth_kaband 4 bytes Array: nbasepoint x nscan
- spacecraft_lat 4 bytes Array: nscan
- spacecraft_lon 4 bytes Array: nscan
- spacecraft_alt 4 bytes Array: nscan
- sun_vector 4 bytes Array: nsunvec x nscan
- sun_elevation_angle 4 bytes Array: nscan
- sun_azimuth_angle 4 bytes Array: nscan
- solar_beta_angle 4 bytes Array: nscan
- time_since_eclipse 4 bytes Array: nscan
- eclipse_status 1 byte Array: nscan
- phase_orbit_midnight 4 bytes Array: nscan
- sun_earth_separation 4 bytes Array: nscan
- phase_eclipse_exit 4 bytes Array: nscan
- earth_angle_radius 4 bytes Array: nscan
- temp_arm 4 bytes Array: nscan
- temp_reflector 4 bytes Array: nscan
- original_gain 4 bytes Array: nchannel x nscan
- corrected_gain 4 bytes Array: nchannel x nscan
- delta_gain 4 bytes Array: nchannel x nscan
- lat_img1 4 bytes Array: npixel_img x nscan

continued on next figure

Figure 230: Data Format Structure for 1BASESSMIS, SSMIS base
continued from last figure

Figure 231: Data Format Structure for 1BASESSMIS, SSMIS base
continued from last figure

- **quality_env2** 1 byte, Array: npixel_env x nscan
- **tb37v_env2** 4 bytes, Array: npixel_env x nscan
- **tb37h_env2** 4 bytes, Array: npixel_env x nscan
- **sun_glint_img1** 4 bytes, Array: npixel_img x nscan
- **sun_glint_env1** 4 bytes, Array: npixel_env x nscan
- **sun_glint_img2** 4 bytes, Array: npixel_img x nscan
- **sun_glint_env2** 4 bytes, Array: npixel_env x nscan
- **sc_azimuth_img** 4 bytes, Array: npixel_img x nscan
- **sc_azimuth_env** 4 bytes, Array: npixel_env x nscan
- **sun_zenith_img** 4 bytes, Array: npixel_img x nscan
- **sun_zenith_env** 4 bytes, Array: npixel_env x nscan
- **sun_azimuth_img** 4 bytes, Array: npixel_img x nscan
- **sun_azimuth_env** 4 bytes, Array: npixel_env x nscan

Figure 232: Data Format Structure for 1BASESSMIS, SSMIS base
Figure 233: Data Format Structure for 1BASESSMIS, baseHeader

Figure 234: Data Format Structure for 1BASESSMIS, ScanTime
InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

baseHeader (Group)

spacecraft_id (4-byte integer, array size: 1):
Satellite ID number.

ascend_time (4-byte integer, array size: 6):
Ascending time.

TLE_xtime (8-byte float, array size: 1):
TLE time.

TLE_date (4-byte integer, array size: 6):
TLE date time arrays.

TLE_time (8-byte float, array size: 1):
TLE time as in two line element.

status_flags1 (4-byte integer, array size: 1):
Processing status flags 1.

status_flags2 (4-byte integer, array size: 1):
Processing status flags 2.

quality_tests (4-byte integer, array size: 6):
Results from quality control tests.

nominal_elevation_angle (4-byte float, array size: 1):
Nominal sensor elevation angle.

spacecraft_roll (4-byte float, array size: 1):
Spacecraft roll angle offset from nominal.

spacecraft_pitch (4-byte float, array size: 1):
Spacecraft pitch angle offset from nominal.
**spacecraft_yaw** (4-byte float, array size: 1):
Spacecraft yaw angle offset from nominal.

**delta_elevation_angle** (4-byte float, array size: nsensor):
Offset in the sensor elevation angle from nominal.

**sensor_roll** (4-byte float, array size: nsensor):
Sensor offset from spacecraft roll angle for each of the six feedhorns (env1, env2, img1, img2, las, uas).

**sensor_pitch** (4-byte float, array size: nsensor):
Sensor offset from spacecraft pitch angle for each of the six feedhorns (env1, env2, img1, img2, las, uas).

**sensor_yaw** (4-byte float, array size: nsensor):
Sensor offset from spacecraft yaw angle for each of the six feedhorns (env1, env2, img1, img2, las, uas).

### S1 (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
5.9 1BASESSMIS - SSMIS base

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

isMissing (1-byte integer, array size: nscan):
Missing scan flag.

scan_time (4-byte integer, array size: ntime x nscan):
Scan start time.

orbit (8-byte float, array size: nscan):
Fractional orbit number.

xtime (8-byte float, array size: nscan):
scan time, seconds since 1987-01-01 00:00:00.

scposx_gci (8-byte float, array size: nsatpos x nscan):
Orbital Position Vector X in Geocentric Inertial Coordinates.

scposy_gci (8-byte float, array size: nsatpos x nscan):
Orbital Position Vector Y in Geocentric Inertial Coordinates.

scposz_gci (8-byte float, array size: nsatpos x nscan):
Orbital Position Vector Z in Geocentric Inertial Coordinates.

scvelx_gci (8-byte float, array size: nsatpos x nscan):
Orbital Velocity Vector X in Geocentric Inertial Coordinates.

scvely_gci (8-byte float, array size: nsatpos x nscan):
Orbital Velocity Vector Y in Geocentric Inertial Coordinates.

scvelz_gci (8-byte float, array size: nsatpos x nscan):
Orbital Velocity Vector Z in Geocentric Inertial Coordinates.

scjulday_orig (4-byte integer, array size: nephem x nscan):
Original spacecraft julian day from TDR data.
sctime_orig (4-byte float, array size: nephem x nscan):
Original spacecraft time from TDR data.

sclat_orig (4-byte float, array size: nephem x nscan):
Original spacecraft latitude from TDR data.

sclon_orig (4-byte float, array size: nephem x nscan):
Original spacecraft longitude from TDR data.

scalt_orig (4-byte float, array size: nephem x nscan):
Original spacecraft altitude from TDR data.

surface_tag_img (1-byte integer, array size: npixel_img x nscan):
Surface tag for imager channels.

rain_flag_img (1-byte integer, array size: npixel_img x nscan):
Rain flag (-1=indeterminate, 0=no rain, 1=rain).

lat_img1_orig (4-byte float, array size: npixel_img x nscan):
Original pixel latitude for channels 150, 183 GHz.

lon_img1_orig (4-byte float, array size: npixel_img x nscan):
Original pixel longitude for channels 150, 183 GHz.

ta150h_img1 (4-byte float, array size: npixel_img x nscan):
150.0 GHz H-Pol Antenna Temperature.

ta183_1h_img1 (4-byte float, array size: npixel_img x nscan):
183.31 +/- 1 GHz H-Pol Antenna Temperature.

ta183_3h_img1 (4-byte float, array size: npixel_img x nscan):
183.31 +/- 3 GHz H-Pol Antenna Temperature.

ta183_7h_img1 (4-byte float, array size: npixel_img x nscan):
183.31 +/- 6.6 GHz H-Pol Antenna Temperature.

lat_img2_orig (4-byte float, array size: npixel_img x nscan):
Original pixel latitude for channels 91 GHz.

lon_img2_orig (4-byte float, array size: npixel_img x nscan):
Original pixel longitude for channels 91 GHz.

ta91v_img2 (4-byte float, array size: npixel_img x nscan):
91.655 GHz V-Pol Antenna Temperature.

ta91h_img2 (4-byte float, array size: npixel_img x nscan):
91.655 GHz H-Pol Antenna Temperature.

surface_tag_env (1-byte integer, array size: npixel_env x nscan):
Surface tag for environmental scene channels.

lat_env1_orig (4-byte float, array size: npixel_env x nscan):
Original pixel latitude for channels 19, 22 GHz.

lon_env1_orig (4-byte float, array size: npixel_env x nscan):
Original pixel longitude for channels 19, 22 GHz.
**ta19v.env1** (4-byte float, array size: npixel_env x nscan): 19.35 GHz V-Pol Antenna Temperature.

**ta19h.env1** (4-byte float, array size: npixel_env x nscan): 19.35 GHz H-Pol Antenna Temperature.

**ta22v.env1** (4-byte float, array size: npixel_env x nscan): 22.235 GHz V-Pol Antenna Temperature.

**lat.env2.orig** (4-byte float, array size: npixel_env x nscan): Original pixel latitude for channels 37 GHz.

**lon.env2.orig** (4-byte float, array size: npixel_env x nscan): Original pixel longitude for channels 37 GHz.

**ta37v.env2** (4-byte float, array size: npixel_env x nscan): 37.0 GHz V-Pol Antenna Temperature.

**ta37h.env2** (4-byte float, array size: npixel_env x nscan): 37.0 GHz H-Pol Antenna Temperature.

**aux.warmcal** (2-byte integer, array size: nchannel x nscan): Warm load calibration by channel (1-24).

**aux.coldcal** (2-byte integer, array size: nchannel x nscan): Cold load calibration by channel (1-24).

**aux.warmloadtemp** (4-byte float, array size: nwarmload x nscan): Warm load temperatures (1-3).

**auxMUXsubID** (2-byte integer, array size: nscan): Subframe ID number.

**auxMUXhouse** (4-byte float, array size: nmuxhouse x nscan): MUX housekeeping values (1-4).

**aux.lat.kband** (4-byte float, array size: nbasepoint x nscan): K-Band Base Point latitude (1-28).

**aux.lon.kband** (4-byte float, array size: nbasepoint x nscan): K-Band Base Point longitude (1-28).

**aux.eia.kband** (4-byte float, array size: nbasepoint x nscan): K-Band Base Point EIA (1-28).

**aux.azimuth.kband** (4-byte float, array size: nbasepoint x nscan): K-Band Base Point Azimuth (1-28).

**aux.lat.uvband** (4-byte float, array size: nbasepoint x nscan): UV-Band Base Point latitude.

**aux.lon.uvband** (4-byte float, array size: nbasepoint x nscan): UV-Band Base Point longitude.

**aux.eia.uvband** (4-byte float, array size: nbasepoint x nscan): UV-Band Base Point EIA.
aux_azimuth_uvb (4-byte float, array size: nbasepoint x nscan):
UV-Band Base Point azimuth.

aux_lat_w (4-byte float, array size: nbasepoint x nscan):
W-Band Base Point latitude.

aux_lon_w (4-byte float, array size: nbasepoint x nscan):
W-Band Base Point longitude.

aux_eia_w (4-byte float, array size: nbasepoint x nscan):
W-Band Base Point EIA.

aux_azimuth_w (4-byte float, array size: nbasepoint x nscan):
W-Band Base Point azimuth.

aux_lat_g (4-byte float, array size: nbasepoint x nscan):
G-Band Base Point latitude.

aux_lon_g (4-byte float, array size: nbasepoint x nscan):
G-Band Base Point longitude.

aux_eia_g (4-byte float, array size: nbasepoint x nscan):
G-Band Base Point EIA.

aux_azimuth_g (4-byte float, array size: nbasepoint x nscan):
G-Band Base Point azimuth.

aux_lat_lv (4-byte float, array size: nbasepoint x nscan):
LV-Band Base Point latitude.

aux_lon_lv (4-byte float, array size: nbasepoint x nscan):
LV-Band Base Point longitude.

aux_eia_lv (4-byte float, array size: nbasepoint x nscan):
LV-Band Base Point EIA.

aux_azimuth_lv (4-byte float, array size: nbasepoint x nscan):
LV-Band Base Point azimuth.

aux_lat_k (4-byte float, array size: nbasepoint x nscan):
Ka-Band Base Point latitude.

aux_lon_k (4-byte float, array size: nbasepoint x nscan):
Ka-Band Base Point longitude.

aux_eia_k (4-byte float, array size: nbasepoint x nscan):
Ka-Band Base Point EIA.

aux_azimuth_k (4-byte float, array size: nbasepoint x nscan):
Ka-Band Base Point azimuth.

spacecraft_lat (4-byte float, array size: nscan):
Spacecraft latitude.

spacecraft_lon (4-byte float, array size: nscan):
Spacecraft longitude.
spacecraft_alt (4-byte float, array size: nscan):
Spacecraft altitude.
sun_vector (4-byte float, array size: nsunvec x nscan):
Sun vector from spacecraft in GCI coordinates.
sun_elevation_angle (4-byte float, array size: nscan):
Sun elevation angle from spacecraft.
sun_azimuth_angle (4-byte float, array size: nscan):
Sun azimuth angle from spacecraft.
solar_beta_angle (4-byte float, array size: nscan):
Solar beta angle.
time_since_eclipse (4-byte float, array size: nscan):
Time since eclipse.
eclipse_status (1-byte integer, array size: nscan):
Eclipse status (0=sun, 1=shadow).

phase_orbit_midnight (4-byte float, array size: nscan):
Phase from orbit midnight.
sun_earth_separation (4-byte float, array size: nscan):
Sun Earth separation angle.

phase_eclipse_exit (4-byte float, array size: nscan):
Orbit phase for eclipse exit or entry.
earth_angle_radius (4-byte float, array size: nscan):
Earth angle radius.
temp_arm (4-byte float, array size: nscan):
Temperature of the reflector arm (measured).
temp_reflector (4-byte float, array size: nscan):
Temperature of the main reflector (estimated from reflector arm temp).
original_gain (4-byte float, array size: nchannel x nscan):
Original smoothed gain.
corrected_gain (4-byte float, array size: nchannel x nscan):
Smoothed gain corrected for solar intrusion.
delta_gain (4-byte float, array size: nchannel x nscan):
Delta gain change for solar intrusion correction.
lat_img1 (4-byte float, array size: npixel_img x nscan):
Pixel latitude for channels 8-11 (150, 183 GHz).
lon_img1 (4-byte float, array size: npixel_img x nscan):
Pixel longitude for channels 8-11 (150, 183 GHz).
eia_img1 (4-byte float, array size: npixel_img x nscan):
Earth Incidence Angle for imager scene channels (150 and 183 +/- 1,3,7).
**quality_img1** (1-byte integer, array size: npixel_img x nscan):
Quality flags for imager scene channels (150 and 183 +/- 1,3,7). Flag definition:

0  Good Data
1  Possible sun glint
2  Climatology check warning (19V Channel)
3  Climatology check warning (19H Channel)
4  Climatology check warning (22V Channel)
5  Climatology check warning (37V Channel)
6  Climatology check warning (37H Channel)
7  Climatology check warning (91V Channel)
8  Climatology check warning (91H Channel)
9  Climatology check warning (150H Channel)
10 Climatology check warning (183+/-1 Channel)
11 Climatology check warning (183+/-3 Channel)
12 Climatology check warning (183+/-7 Channel)
13 Climatology check warning (Multiple enviro sensor channels)
14 Climatology check warning (Multiple imager sensor channels)
15 Climatology check warning (Multiple LAS sensor channels)
16 Climatology check warning (Multiple UAS sensor channels)
17 Correction for lunar intrusion into warm load
18 Correction for solar intrusion into warm load
19 No sun angle correction warning in multiple channels
20 Sensor data issue in multiple imager sensor channels
21 Sensor data issue in multiple enviro sensor channels
22 Sensor data issue in 91H channel
101 Geolocation check flagged in input BASE file
102 Climatology check flagged in input BASE file
103 Antenna temperatures are less than 50 or greater than 350
110 Climatology check failure (19V Channel)
111 Climatology check failure (19H Channel)
112 Climatology check failure (22V Channel)
113 Climatology check failure (37V Channel)
114 Climatology check failure (37H Channel)
115 Climatology check failure (91V Channel)
116 Climatology check failure (91H Channel)
117 Climatology check failure (150H Channel)
118 Climatology check failure (183+/-1 Channel)
119 Climatology check failure (183+/-3 Channel)
120 Climatology check failure (183+/-7 Channel)
121 Climatology check failure (Multiple enviro sensor channels)
122 Climatology check failure (Multiple imager sensor channels)
123 Climatology check failure (Multiple LAS sensor channels)
124 Climatology check failure (Multiple UAS sensor channels)
5.9 1BASESSMIS - SSMIS base

125 Failure of 150H channel
126 Failure of multiple imager sensor channel
127 Failure of 37V channel

\textbf{tb150h\_img1} (4-byte float, array size: npixel\_img x nscan): 150.0 GHz H-Pol Brightness Temperature.

\textbf{tb183\_1h\_img1} (4-byte float, array size: npixel\_img x nscan): 183.31 +/- 1 GHz H-Pol Brightness Temperature.

\textbf{tb183\_3h\_img1} (4-byte float, array size: npixel\_img x nscan): 183.31 +/- 3 GHz H-Pol Brightness Temperature.

\textbf{tb183\_7h\_img1} (4-byte float, array size: npixel\_img x nscan): 183.31 +/- 6.6 GHz H-Pol Brightness Temperature.

\textbf{lat\_img2} (4-byte float, array size: npixel\_img x nscan): Pixel latitude for channels 17-18 (91 GHz).

\textbf{lon\_img2} (4-byte float, array size: npixel\_img x nscan): Pixel longitude for channels 17-18 (91 GHz).

\textbf{eia\_img2} (4-byte float, array size: npixel\_img x nscan): Earth Incidence Angle for imager scene channels (91v and 91h).

\textbf{quality\_img2} (1-byte integer, array size: npixel\_img x nscan): Quality flags for imager scene channels (91v and 91h).

\textbf{tb91v\_img2} (4-byte float, array size: npixel\_img x nscan): 19.35 GHz V-Pol Brightness Temperature.

\textbf{tb91h\_img2} (4-byte float, array size: npixel\_img x nscan): 19.35 GHz H-Pol Brightness Temperature.

\textbf{lat\_env1} (4-byte float, array size: npixel\_env x nscan): Pixel latitude for channels 12-14 (19, 22 GHz).

\textbf{lon\_env1} (4-byte float, array size: npixel\_env x nscan): Pixel longitude for channels 12-14 (19, 22 GHz).

\textbf{eia\_env1} (4-byte float, array size: npixel\_env x nscan): Earth Incidence Angle for environmental scene channels (19v, 19h, and 22v).

\textbf{quality\_env1} (1-byte integer, array size: npixel\_env x nscan): Quality flags for environmental scene channels (19v, 19h, and 22v).

\textbf{tb19v\_env1} (4-byte float, array size: npixel\_env x nscan): 19.35 GHz V-Pol Brightness Temperature.

\textbf{tb19h\_env1} (4-byte float, array size: npixel\_env x nscan): 19.35 GHz H-Pol Brightness Temperature.

\textbf{tb22v\_env1} (4-byte float, array size: npixel\_env x nscan): 22.235 GHz V-Pol Brightness Temperature.
lat_env2 (4-byte float, array size: npixel_env x nscan):
Pixel latitude for channels 15-16 (37 GHz).

lon_env2 (4-byte float, array size: npixel_env x nscan):
Pixel longitude for channels 15-16 (37 GHz).

eia_env2 (4-byte float, array size: npixel_env x nscan):
Earth Incidence Angle for environmental scene channels (37v and 37h).

quality_env2 (1-byte integer, array size: npixel_env x nscan):
Quality flags for environmental scene channels (37v and 37h).

tb37v_env2 (4-byte float, array size: npixel_env x nscan):
37.0 GHz V-Pol Brightness Temperature.

tb37h_env2 (4-byte float, array size: npixel_env x nscan):
37.0 GHz H-Pol Brightness Temperature.

sun_glint_img1 (4-byte float, array size: npixel_img x nscan):
Sun glint angle for channels 8-11 (150, 183 GHz).

sun_glint_env1 (4-byte float, array size: npixel_env x nscan):
Sun glint angle for channels 12-14 (19, 22 GHz).

sun_glint_img2 (4-byte float, array size: npixel_img x nscan):
Sun glint angle for channels 17-18 (91 GHz).

sun_glint_env2 (4-byte float, array size: npixel_env x nscan):
Sun glint angle for channels 15-16 (37 GHz)

sc_azimuth_img (4-byte float, array size: npixel_img x nscan):
Satellite azimuth angle for channels 8-11, 17-18 (91, 150, 183 GHz).

sc_azimuth_env (4-byte float, array size: npixel_env x nscan):
Satellite azimuth angle for channels 12-16 (19, 22, 37 GHz).

sun_zenith_img (4-byte float, array size: npixel_img x nscan):
Sun zenith angle for channels 8-11, 17-18 (91, 150, 183 GHz).

sun_zenith_env (4-byte float, array size: npixel_env x nscan):
Sun zenith angle for channels 12-16 (19, 22, 37 GHz).

sun_azimuth_img (4-byte float, array size: npixel_img x nscan):
Sun azimuth angle for channels 8-11, 17-18 (91, 150, 183 GHz).

sun_azimuth_env (4-byte float, array size: npixel_env x nscan):
Sun azimuth angle for channels 12-16 (19, 22, 37 GHz).

C Structure Header file:

```c
#ifndef _TK_1BASESSMIS_H_
#define _TK_1BASESSMIS_H_

#ifndef _SCANTIME_
```

#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L1BASESSMIS_S1_
#define _L1BASESSMIS_S1_

typedef struct {
    SCANTIME ScanTime;
    signed char isMissing;
    int scan_time[7];
    double orbit;
    double xtime;
    double scposx_gci[2];
    double scposy_gci[2];
    double scposz_gci[2];
    double scvelx_gci[2];
    double scvely_gci[2];
    double scvelz_gci[2];
    int scjulday_orig[3];
    float sctime_orig[3];
    float sclat_orig[3];
    float sclon_orig[3];
    float scalt_orig[3];
    signed char surface_tag_img[180];
    signed char rain_flag_img[180];
    float lat_img1_orig[180];
    float lon_img1 Orig[180];
    float ta150h_img1[180];
    float ta183_1h_img1[180];
    float ta183_3h_img1[180];
} L1BASESSMIS_S1;

#endif
float ta183_7h_img1[180];
float lat_img2_orig[180];
float lon_img2_orig[180];
float ta91v_img2[180];
float ta91h_img2[180];
signed char surface_tag_env[90];
float lat_env1_orig[90];
float lon_env1_orig[90];
float ta19v_env1[90];
float ta19h_env1[90];
float ta22v_env1[90];
float lat_env2_orig[90];
float lon_env2_orig[90];
float ta37v_env2[90];
float ta37h_env2[90];
short aux_warmcal[24];
short aux_coldcal[24];
float aux_warmloadtemp[3];
short aux_MUXsubID;
float aux_MUXhouse[4];
float aux_lat_kband[28];
float aux_lon_kband[28];
float aux_eia_kband[28];
float aux_azimuth_kband[28];
float aux_lat_uvbnd[28];
float aux_lon_uvbnd[28];
float aux_eia_uvbnd[28];
float aux_azimuth_uvbnd[28];
float aux_lat_wband[28];
float aux_lon_wband[28];
float aux_eia_wband[28];
float aux_azimuth_wband[28];
float aux_lat_gband[28];
float aux_lon_gband[28];
float aux_eia_gband[28];
float aux_azimuth_gband[28];
float aux_lat_lvband[28];
float aux_lon_lvband[28];
float aux_eia_lvband[28];
float aux_azimuth_lvband[28];
float aux_lat_kaband[28];
float aux_lon_kaband[28];
float aux_eia_kaband[28];
float aux_azimuth_kaband[28];
float spacecraft_lat;
float spacecraft_lon;
float spacecraft_alt;
float sun_vector[3];
float sun_elevation_angle;
float sun_azimuth_angle;
float solar_beta_angle;
float time_since_eclipse;
signed char eclipse_status;
float phase_orbit_midnight;
float sun_earth_separation;
float phase_eclipse_exit;
float earth_angle_radius;
float temp_arm;
float temp_reflector;
float original_gain[24];
float corrected_gain[24];
float delta_gain[24];
float lat_img1[180];
float lon_img1[180];
float eia_img1[180];
signed char quality_img1[180];
float tb150h_img1[180];
float tb183_1h_img1[180];
float tb183_3h_img1[180];
float tb183_7h_img1[180];
float lat_img2[180];
float lon_img2[180];
float eia_img2[180];
signed char quality_img2[180];
float tb91v_img2[180];
float tb91h_img2[180];
float lat_env1[90];
float lon_env1[90];
float eia_env1[90];
signed char quality_env1[90];
float tb19v_env1[90];
float tb19h_env1[90];
float tb22v_env1[90];
float lat_env2[90];
float lon_env2[90];
float eia_env2[90];
signed char quality_env2[90];
float tb37v_env2[90];
float tb37h_env2[90];
float sun_glint_img1[180];
float sun_glint_env1[90];
float sun_glint_img2[180];
float sun_glint_env2[90];
float sc_azimuth_img[180];
float sc_azimuth_env[90];
float sun_zenith_img[180];
float sun_zenith_env[90];
float sun_azimuth_img[180];
float sun_azimuth_env[90];
} L1BASESSMIS_S1;

#endif

#ifndef _L1BASESSMIS_BASEHEADER_
#define _L1BASESSMIS_BASEHEADER_

typedef struct {
    int spacecraft_id[1];
    int ascend_time[6];
    double TLE_xtime[1];
    int TLE_date[6];
    double TLE_time[1];
    int status_flags1[1];
    int status_flags2[1];
    int quality_tests[6];
    float nominal_elevation_angle[1];
    float spacecraft_roll[1];
    float spacecraft_pitch[1];
    float spacecraft_yaw[1];
    float delta_elevation_angle[6];
    float sensor_roll[6];
    float sensor_pitch[6];
    float sensor_yaw[6];
} L1BASESSMIS_BASEHEADER;
#endif

#endif

Fortran Structure Header file:
STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASESSMIS_S1/
    RECORD /SCANTIME/ ScanTime
    BYTE isMissing
    INTEGER*4 scan_time(7)
    REAL*8 orbit
    REAL*8 xtime
    REAL*8 scposx_gci(2)
    REAL*8 scposy_gci(2)
    REAL*8 scposz_gci(2)
    REAL*8 scvelx_gci(2)
    REAL*8 scvely_gci(2)
    REAL*8 scvelz_gci(2)
    INTEGER*4 scjulday_orig(3)
    REAL*4 sctime_orig(3)
    REAL*4 sclat_orig(3)
    REAL*4 sclon_orig(3)
    REAL*4 scalt_orig(3)
    BYTE surface_tag_img(180)
    BYTE rain_flag_img(180)
    REAL*4 lat_img1_orig(180)
    REAL*4 lon_img1_orig(180)
    REAL*4 ta150h_img1(180)
    REAL*4 ta183_1h_img1(180)
    REAL*4 ta183_3h_img1(180)
    REAL*4 ta183_7h_img1(180)
    REAL*4 lat_img2_orig(180)
    REAL*4 lon_img2_orig(180)
    REAL*4 ta91v_img2(180)
    REAL*4 ta91h_img2(180)
    BYTE surface_tag_env(90)
REAL*4 lat_env1_orig(90)  
REAL*4 lon_env1_orig(90)  
REAL*4 ta19v_env1(90)  
REAL*4 ta19h_env1(90)  
REAL*4 ta22v_env1(90)  
REAL*4 lat_env2_orig(90)  
REAL*4 lon_env2_orig(90)  
REAL*4 ta37v_env2(90)  
REAL*4 ta37h_env2(90)  
INTEGER*2 aux_warmcal(24)  
INTEGER*2 aux_coldcal(24)  
REAL*4 aux_warmloadtemp(3)  
INTEGER*2 aux_MUXsubID  
REAL*4 aux_MUXhouse(4)  
REAL*4 aux_lat_kband(28)  
REAL*4 aux_lon_kband(28)  
REAL*4 aux_eia_kband(28)  
REAL*4 aux_azimuth_kband(28)  
REAL*4 aux_lat_uvband(28)  
REAL*4 aux_lon_uvband(28)  
REAL*4 aux_eia_uvband(28)  
REAL*4 aux_azimuth_uvband(28)  
REAL*4 aux_lat_wband(28)  
REAL*4 aux_lon_wband(28)  
REAL*4 aux_eia_wband(28)  
REAL*4 aux_azimuth_wband(28)  
REAL*4 aux_lat_gband(28)  
REAL*4 aux_lon_gband(28)  
REAL*4 aux_eia_gband(28)  
REAL*4 aux_azimuth_gband(28)  
REAL*4 aux_lat_lvband(28)  
REAL*4 aux_lon_lvband(28)  
REAL*4 aux_eia_lvband(28)  
REAL*4 aux_azimuth_lvband(28)  
REAL*4 spacecraft_lat  
REAL*4 spacecraft_lon  
REAL*4 spacecraft_alt  
REAL*4 sun_vector(3)  
REAL*4 sun_elevation_angle
REAL*4 sun_azimuth_angle
REAL*4 solar_beta_angle
REAL*4 time_since_eclipse
BYTE eclipse_status
REAL*4 phase_orbit_midnight
REAL*4 sun_earth_separation
REAL*4 phase_eclipse_exit
REAL*4 earth_angle_radius
REAL*4 temp_arm
REAL*4 temp_reflector
REAL*4 original_gain(24)
REAL*4 corrected_gain(24)
REAL*4 delta_gain(24)
REAL*4 lat_img1(180)
REAL*4 lon_img1(180)
REAL*4 eia_img1(180)
BYTE quality_img1(180)
REAL*4 tb150h_img1(180)
REAL*4 tb183_1h_img1(180)
REAL*4 tb183_3h_img1(180)
REAL*4 tb183_7h_img1(180)
REAL*4 lat_img2(180)
REAL*4 lon_img2(180)
REAL*4 eia_img2(180)
BYTE quality_img2(180)
REAL*4 tb91v_img2(180)
REAL*4 tb91h_img2(180)
REAL*4 lat_env1(90)
REAL*4 lon_env1(90)
REAL*4 eia_env1(90)
BYTE quality_env1(90)
REAL*4 tb19v_env1(90)
REAL*4 tb19h_env1(90)
REAL*4 tb22v_env1(90)
REAL*4 lat_env2(90)
REAL*4 lon_env2(90)
REAL*4 eia_env2(90)
BYTE quality_env2(90)
REAL*4 tb37v_env2(90)
REAL*4 tb37h_env2(90)
REAL*4 sun_glint_img1(180)
REAL*4 sun_glint_env1(90)
REAL*4 sun_glint_img2(180)
REAL*4  sun_glint_env2(90)
REAL*4  sc_azimuth_img(180)
REAL*4  sc_azimuth_env(90)
REAL*4  sun_zenith_img(180)
REAL*4  sun_zenith_env(90)
REAL*4  sun_azimuth_img(180)
REAL*4  sun_azimuth_env(90)

END STRUCTURE

STRUCTURE /L1BASESSMIS_BASEHEADER/
  INTEGER*4  spacecraft_id(1)
  INTEGER*4  ascend_time(6)
  REAL*8    TLE_xtime(1)
  INTEGER*4  TLE_date(6)
  REAL*8    TLE_time(1)
  INTEGER*4  status_flags1(1)
  INTEGER*4  status_flags2(1)
  INTEGER*4  quality_tests(6)
  REAL*4    nominal_elevation_angle(1)
  REAL*4    spacecraft_roll(1)
  REAL*4    spacecraft_pitch(1)
  REAL*4    spacecraft_yaw(1)
  REAL*4    delta_elevation_angle(6)
  REAL*4    sensor_roll(6)
  REAL*4    sensor_pitch(6)
  REAL*4    sensor_yaw(6)

END STRUCTURE

5.10  **1BASEAMSRE - AMSRE base**

1BASEAMSRE contains brightness temperature. These files contain all the information from the AMSR-E Level 1B files produced by JAXA plus the isMissing flag, sun_Glint_Angle, solarBetaAngle and timeSinceEclipseEntry. 1BASEAMSRE is written as one swath, but the 1C product will have 6 swaths. More detailed information on some variables may be found in the document "AMSRE Level 1 Product Format Specification" written by JAXA.

NOTE: the filespec does not draw the swath. NOTE: ranges may be wrong

Dimension definitions:
nscan var Number of scans in the granule.
npixel1 196 Number pixels for low res.
npixel2 392 Number pixels for high res.
count1 16 Number of (hot or cold) load for 6GHz to 52GHz.
count2 32 Number of (hot or cold) load for 89GHz.
freq1 12 Number of freq. for 6GHz to 52GHz.
freq2 4 Number of freq. for 89GHz.

Figure 235 through Figure 239 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**baseHeader** (Group)

**CoRegistrationParameterA1** (4-byte float, array size: 6):
Co-registration parameter A1 used for calculating latitude and longitude of the observing point for each frequency except 89 GHz.

**CoRegistrationParameterA2** (4-byte float, array size: 6):
Co-registration parameter A2 used for calculating latitude and longitude of the observing point for each frequency except 89 GHz.

**S1** (Swath)
Figure 235: Data Format Structure for 1BASEAMSRE, AMSRE base
continued from last figure

<table>
<thead>
<tr>
<th>Variable</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(T_b_{23\text{GHz_V}})</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(T_b_{23\text{GHz_H}})</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(T_b_{36\text{GHz_V}})</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(T_b_{36\text{GHz_H}})</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(T_b_{89\text{GHz_V_A}})</td>
<td>4 bytes</td>
<td>Array: npixel2 x nscan</td>
</tr>
<tr>
<td>(T_b_{89\text{GHz_H_A}})</td>
<td>4 bytes</td>
<td>Array: npixel2 x nscan</td>
</tr>
<tr>
<td>(T_b_{89\text{GHz_V_B}})</td>
<td>4 bytes</td>
<td>Array: npixel2 x nscan</td>
</tr>
<tr>
<td>(T_b_{89\text{GHz_H_B}})</td>
<td>4 bytes</td>
<td>Array: npixel2 x nscan</td>
</tr>
<tr>
<td>(\text{Hot_Load_Count}^{6\to52})</td>
<td>2 bytes</td>
<td>Array: freq1 x count1 x nscan</td>
</tr>
<tr>
<td>(\text{Hot_Load_Count}^{89})</td>
<td>2 bytes</td>
<td>Array: freq2 x count2 x nscan</td>
</tr>
<tr>
<td>(\text{Cold_Sky_Mirror_Count}^{6\to52})</td>
<td>2 bytes</td>
<td>Array: freq1 x count1 x nscan</td>
</tr>
<tr>
<td>(\text{Antenna_Temp_Coef})</td>
<td>4 bytes</td>
<td>Array: 32 x nscan</td>
</tr>
<tr>
<td>(\text{Rx_Offset_Gain_Count})</td>
<td>2 bytes</td>
<td>Array: 32 x nscan</td>
</tr>
<tr>
<td>(\text{Latitude}_6)</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(\text{Longitude}_6)</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(\text{Latitude}_10)</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(\text{Longitude}_10)</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(\text{Latitude}_18)</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(\text{Longitude}_18)</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(\text{Latitude}_23)</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(\text{Longitude}_23)</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
<tr>
<td>(\text{Latitude}_36)</td>
<td>4 bytes</td>
<td>Array: npixel1 x nscan</td>
</tr>
</tbody>
</table>

continued on next figure

Figure 236: Data Format Structure for 1BASEAMSRE, AMSRE base
continued from last figure

- **Longitude** 36 4 bytes Array: npixel1 x nscan
- **Latitude** 89A 4 bytes Array: npixel2 x nscan
- **Longitude** 89A 4 bytes Array: npixel2 x nscan
- **Latitude** 89B 4 bytes Array: npixel2 x nscan
- **Longitude** 89B 4 bytes Array: npixel2 x nscan
- **Earth_Incidence_6** 4 bytes Array: npixel1 x nscan
- **Earth_Incidence_10** 4 bytes Array: npixel1 x nscan
- **Earth_Incidence_18** 4 bytes Array: npixel1 x nscan
- **Earth_Incidence_23** 4 bytes Array: npixel1 x nscan
- **Earth_Incidence_36** 4 bytes Array: npixel1 x nscan
- **Earth_Incidence_89A** 4 bytes Array: npixel2 x nscan
- **Earth_Incidence_89B** 4 bytes Array: npixel2 x nscan
- **Sun_Glint_Angle** 4 bytes Array: npixel1 x nscan
- **Earth_Azimuth** 4 bytes Array: npixel1 x nscan
- **Land_Ocean_Flag_for_6,10,18,23,36,50,89A** 1 byte Array: 7 x npixel1 x nscan
- **Observation_Supplement** 2 bytes Array: 27 x nscan
- **SPC_Temperature_Count** 2 bytes Array: 20 x nscan
- **SPS_Temperature_Count** 2 bytes Array: 32 x nscan
- **Data_Quality** 4 bytes Array: 128 x nscan
- **Interpolation_Flag_6_to_52** 1 byte Array: freq1 x count1 x nscan
- **Interpolation_Flag_89** 1 byte Array: freq2 x count2 x nscan
- **Spill_Over** 4 bytes Array: 2 x 243 x nscan

Figure 237: Data Format Structure for 1BASEAMSRE, AMSRE base

- **CoRegistrationParameterA1** 4 bytes Array: 6
- **CoRegistrationParameterA2** 4 bytes Array: 6

Figure 238: Data Format Structure for 1BASEAMSRE, baseHeader
SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.
Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value
Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value
DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value
Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value
Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value
millisecond (2-byte integer, array size: nscan):
Thousands of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

dayofyear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

secondofday (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

ismissing (1-byte integer, array size: nscan):
Missing scan flag.
solarbetaangle (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given
by the cross product of the spacecraft position and velocity vectors. Values range from
-59.0 to 59.0 degrees. Special values are defined as:
-9999.9 Missing value
timesinceeclipseentry (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow.
sclat (4-byte float, array size: nscan):
Spacecraft latitude. Values range from -90 to 90.0 degree. Special values are defined as:
-9999.9 Missing value

celon (4-byte float, array size: nscan):
Spacecraft longitude. Values range from -180 to 180.0 degree. Special values are defined
as:
-9999.9 Missing value

calt (4-byte float, array size: nscan):
Spacecraft altitude. Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

amsre_scan_time (8-byte float, array size: nscan):
The observation start time of 89GHz A-horn. This time is a total second (TAI) from
00:00 (UTC) on January 1st, 1993.

position_in_orbit (8-byte float, array size: nscan):
Fractional orbit number.

navigation_data (4-byte float, array size: 6 x nscan):
Satellite position and velocity corresponding to the observation start time (Amsre.Scan_Time)
of each scan. Data is in earth-centered inertial (ECI), J2000 coordinates.

attitude_data (4-byte float, array size: 3 x nscan):
Roll, Pitch and Yaw corresponding to the observation start time of each scan.
\textbf{Tb}_{6\text{GHz}}_V$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 6 GHz vertical polarization.

\textbf{Tb}_{6\text{GHz}}_H$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 6 GHz horizontal polarization.

\textbf{Tb}_{10\text{GHz}}_V$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 10 GHz vertical polarization.

\textbf{Tb}_{10\text{GHz}}_H$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 10 GHz horizontal polarization.

\textbf{Tb}_{18\text{GHz}}_V$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 18 GHz vertical polarization.

\textbf{Tb}_{18\text{GHz}}_H$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 18 GHz horizontal polarization.

\textbf{Tb}_{23\text{GHz}}_V$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 23 GHz vertical polarization.

\textbf{Tb}_{23\text{GHz}}_H$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 23 GHz horizontal polarization.

\textbf{Tb}_{36\text{GHz}}_V$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 36 GHz vertical polarization.

\textbf{Tb}_{36\text{GHz}}_H$ (4-byte float, array size: npixel1 x nscan): Brightness temperature of 36 GHz horizontal polarization.

\textbf{Tb}_{89\text{GHz}}_V\_A$ (4-byte float, array size: npixel2 x nscan): Brightness temperature of 89 GHz A-horns vertical polarization.

\textbf{Tb}_{89\text{GHz}}_H\_A$ (4-byte float, array size: npixel2 x nscan): Brightness temperature of 89 GHz A-horns horizontal polarization.

\textbf{Tb}_{89\text{GHz}}_V\_B$ (4-byte float, array size: npixel2 x nscan): Brightness temperature of 89 GHz B-horns vertical polarization.

\textbf{Tb}_{89\text{GHz}}_H\_B$ (4-byte float, array size: npixel2 x nscan): Brightness temperature of 89 GHz B-horns horizontal polarization.

\textbf{Hot\_Load\_Count\_6\_to\_52}$ (2-byte integer, array size: freq1 x count1 x nscan): The observed count of HTS and polarization for each frequency except 89 GHz.

\textbf{Hot\_Load\_Count\_89}$ (2-byte integer, array size: freq2 x count2 x nscan): The observed count of HTS and polarization for 89 GHz.

\textbf{Cold\_Sky\_Mirror\_Count\_6\_to\_52}$ (2-byte integer, array size: freq1 x count1 x nscan): The observed count of CSM and polarization for each frequency except 89 GHz.

\textbf{Cold\_Sky\_Mirror\_Count\_89}$ (2-byte integer, array size: freq2 x count2 x nscan): The observed count of CSM and polarization for 89 GHz.
**Antenna_Temp_Coef** (4-byte float, array size: 32 x nscan): The antenna temperature conversion coefficients used for converting the observed count value into antenna temperature.

**Rx_Offset_Gain_Count** (2-byte unsigned integer, array size: 32 x nscan): The gain and offset values for receiver (RX) of each frequency.

**Latitude_6** (4-byte float, array size: npixel1 x nscan): Latitude of the observation point on the earth surface at 6GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.

**Longitude_6** (4-byte float, array size: npixel1 x nscan): Longitude of the observation point on the earth surface at 6GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.

**Latitude_10** (4-byte float, array size: npixel1 x nscan): Latitude of the observation point on the earth surface at 10GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.

**Longitude_10** (4-byte float, array size: npixel1 x nscan): Longitude of the observation point on the earth surface at 10GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.

**Latitude_18** (4-byte float, array size: npixel1 x nscan): Latitude of the observation point on the earth surface at 18GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.

**Longitude_18** (4-byte float, array size: npixel1 x nscan): Longitude of the observation point on the earth surface at 18GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.

**Latitude_23** (4-byte float, array size: npixel1 x nscan): Latitude of the observation point on the earth surface at 23GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.

**Longitude_23** (4-byte float, array size: npixel1 x nscan): Longitude of the observation point on the earth surface at 23GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.

**Latitude_36** (4-byte float, array size: npixel1 x nscan): Latitude of the observation point on the earth surface at 36GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.
**Longitude.36** (4-byte float, array size: npixel1 x nscan):
Longitude of the observation point on the earth surface at 36GHz channels. Values are calculated from the observation position of 89 GHz A horn and the co-registration parameters.

**Latitude.89A** (4-byte float, array size: npixel2 x nscan):
Latitude of the observation point on the earth surface at 89GHz A-horn.

**Longitude.89A** (4-byte float, array size: npixel2 x nscan):
Longitude of the observation point on the earth surface at 89GHz A-horn.

**Latitude.89B** (4-byte float, array size: npixel2 x nscan):
Latitude of the observation point on the earth surface at 89GHz B-horn.

**Longitude.89B** (4-byte float, array size: npixel2 x nscan):
Longitude of the observation point on the earth surface at 89GHz B-horn.

**Earth_Incidence.6** (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 6 GHz.

**Earth_Incidence.10** (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 10 GHz.

**Earth_Incidence.18** (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 18 GHz.

**Earth_Incidence.23** (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 23 GHz.

**Earth_Incidence.36** (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 36 GHz.

**Earth_Incidence.89A** (4-byte float, array size: npixel2 x nscan):
The earth incidence angle 89 GHz A.

**Earth_Incidence.89B** (4-byte float, array size: npixel2 x nscan):
The earth incidence angle 89 GHz B.

**Sun_Glint.Angle** (4-byte float, array size: npixel1 x nscan):
The sun glint angle on odd observation points (origin 1) of 89 GHz A-horn.

**Earth_Azimuth** (4-byte float, array size: npixel1 x nscan):
The earth azimuth angle on odd observation points (origin 1) of 89 GHz A-horn.

**Land_Ocean_Flag_for.6.10.18.23.36.50.89A** (1-byte char, array size: 7 x npixel1 x nscan):
The land coverage percentage of the observation footprint.

**Observation.Supplement** (2-byte unsigned integer, array size: 27 x nscan):
Observation supplement raw data such as a H/W state.

**SPC.Temperature.Count** (2-byte integer, array size: 20 x nscan):
The temperature of SPC.
SPS_Temperature_Count (2-byte integer, array size: 32 x nscan):
The temperature of SPS.

Data_Quality (4-byte float, array size: 128 x nscan):
The quality and supplementary information. Please refer to "AMSRE-E Level 1 product format description" for details.

Interpolation_Flag_6_to_52 (1-byte char, array size: freq1 x count1 x nscan):
The interpolation flag for CSM data.

Interpolation_Flag_89 (1-byte char, array size: freq2 x count2 x nscan):
The interpolation flag for CSM data.

Spill_Over (4-byte float, array size: 2 x 243 x nscan):
Spill over.

C Structure Header file:

```c
#ifndef _TK_1BASEAMSRE_H_
define _TK_1BASEAMSRE_H_

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L1BASEAMSRE_S1_
define _L1BASEAMSRE_S1_

typedef struct {
    SCANTIME ScanTime;
    signed char isMissing;
    float solarBetaAngle;
    float timeSinceEclipseEntry;
```
float sclat;
float sclon;
float scalt;
double Amsre_Scan_Time;
double Position_in_Orbit;
float Navigation_Data[6];
float Attitude_Data[3];
float Tb_6GHz_V[196];
float Tb_6GHz_H[196];
float Tb_10GHz_V[196];
float Tb_10GHz_H[196];
float Tb_18GHz_V[196];
float Tb_18GHz_H[196];
float Tb_23GHz_V[196];
float Tb_23GHz_H[196];
float Tb_36GHz_V[196];
float Tb_36GHz_H[196];
float Tb_89GHz_V_A[392];
float Tb_89GHz_H_A[392];
float Tb_89GHz_V_B[392];
float Tb_89GHz_H_B[392];
short Hot_Load_Count_6_to_52[16][12];
short Hot_Load_Count_89[32][4];
short Cold_Sky_Mirror_Count_6_to_52[16][12];
short Cold_Sky_Mirror_Count_89[32][4];
float Antenna_Temp_Coef[32];
unsigned short Rx_Offset_Gain_Count[32];
float Latitude_6[196];
float Longitude_6[196];
float Latitude_10[196];
float Longitude_10[196];
float Latitude_18[196];
float Longitude_18[196];
float Latitude_23[196];
float Longitude_23[196];
float Latitude_36[196];
float Longitude_36[196];
float Latitude_89A[392];
float Longitude_89A[392];
float Latitude_89B[392];
float Longitude_89B[392];
float Earth_Incidence_6[196];
float Earth_Incidence_10[196];
float Earth_Incidence_18[196];
float Earth_Incidence_23[196];
float Earth_Incidence_36[196];
float Earth_Incidence_89A[392];
float Earth_Incidence_89B[392];
float Sun_Glint_Angle[196];
float Earth_Azimuth[196];
unsigned char Land_Ocean_Flag_for_6_10_18_23_36_50_89A[196][7];
unsigned short Observation_Supplement[27];
short SPC_Temperature_Count[20];
short SPS_Temperature_Count[32];
float Data_Quality[128];
unsigned char Interpolation_Flag_6_to_52[16][12];
unsigned char Interpolation_Flag_89[32][4];
float Spill_Over[243][2];
} L1BASEAMSRE_S1;

#endif

#ifndef _L1BASEAMSRE_BASEHEADER_
#define _L1BASEAMSRE_BASEHEADER_

typedef struct {
    float CoRegistrationParameterA1[6];
    float CoRegistrationParameterA2[6];
} L1BASEAMSRE_BASEHEADER;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay

END STRUCTURE

STRUCTURE /L1BASEAMSRE_S1/
  RECORD /SCANTIME/ ScanTime
  BYTE isMissing
  REAL*4 solarBetaAngle
  REAL*4 timeSinceEclipseEntry
  REAL*4 sclat
  REAL*4 sclon
  REAL*4 scalt
  REAL*8 Amsre_Scan_Time
  REAL*8 Position_in_Orbit
  REAL*4 Navigation_Data(6)
  REAL*4 Attitude_Data(3)
  REAL*4 Tb_6GHz_V(196)
  REAL*4 Tb_6GHz_H(196)
  REAL*4 Tb_10GHz_V(196)
  REAL*4 Tb_10GHz_H(196)
  REAL*4 Tb_18GHz_V(196)
  REAL*4 Tb_18GHz_H(196)
  REAL*4 Tb_23GHz_V(196)
  REAL*4 Tb_23GHz_H(196)
  REAL*4 Tb_36GHz_V(196)
  REAL*4 Tb_36GHz_H(196)
  REAL*4 Tb_89GHz_V_A(392)
  REAL*4 Tb_89GHz_H_A(392)
  REAL*4 Tb_89GHz_V_B(392)
  REAL*4 Tb_89GHz_H_B(392)
  INTEGER*2 Hot_Load_Count_6_to_52(12,16)
  INTEGER*2 Hot_Load_Count_89(4,32)
  INTEGER*2 Cold_Sky_Mirror_Count_6_to_52(12,16)
  INTEGER*2 Cold_Sky_Mirror_Count_89(4,32)
  REAL*4 Antenna_Temp_Coef(32)
  INTEGER*2 Rx_Offset_Gain_Count(32)
  REAL*4 Latitude_6(196)
  REAL*4 Longitude_6(196)
  REAL*4 Latitude_10(196)
  REAL*4 Longitude_10(196)
  REAL*4 Latitude_18(196)
  REAL*4 Longitude_18(196)
  REAL*4 Latitude_23(196)
  REAL*4 Longitude_23(196)
  REAL*4 Latitude_36(196)
REAL*4 Longitude_36(196)
REAL*4 Latitude_89A(392)
REAL*4 Longitude_89A(392)
REAL*4 Latitude_89B(392)
REAL*4 Longitude_89B(392)
REAL*4 Earth_Incidence_6(196)
REAL*4 Earth_Incidence_10(196)
REAL*4 Earth_Incidence_18(196)
REAL*4 Earth_Incidence_23(196)
REAL*4 Earth_Incidence_36(196)
REAL*4 Earth_Incidence_89A(392)
REAL*4 Earth_Incidence_89B(392)
REAL*4 Sun_Glint_Angle(196)
REAL*4 Earth_Azimuth(196)
CHARACTER Land_Ocean_Flag_for_6_10_18_23_36_50_89A(7,196)
INTEGER*2 Observation_Supplement(27)
INTEGER*2 SPC_Temperature_Count(20)
INTEGER*2 SPS_Temperature_Count(32)
REAL*4 Data_Quality(128)
CHARACTER Interpolation_Flag_6_to_52(12,16)
CHARACTER Interpolation_Flag_89(4,32)
REAL*4 Spill_Over(2,243)
END STRUCTURE

STRUCTURE /L1BASEAMSRE_BASEHEADER/
  REAL*4 CoRegistrationParameterA1(6)
  REAL*4 CoRegistrationParameterA2(6)
END STRUCTURE

5.11 1BASEAMSR2 - AMSR2 base

1BASEAMSR2 contains brightness temperature. 1BASEAMSR2 is written as one swath, but the 1C product will have 6 swaths. More detailed information on some variables may be found in the document

Global Change Observation Mission Water (GCOM-W1)
AMSR2 Level 1 Product Format Specification
written by JAXA

Dimension definitions:
nscan var Number of scans in the granule.
npixel1 243 Number pixels for low res.
npixel2 486 Number pixels for high res.
count1 16 Number of (hot or cold) load for 6GHz to 36GHz.
count2 32 Number of (hot or cold) load for 89GHz.
freq1 12 Number of freq. for 6GHz to 36GHz.
freq2 4 Number of freq. for 89GHz.
nchanRFI 8 Number of channels for rfiFlag.

Figure 240 through Figure 245 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**baseHeader** (Group)

**CoRegistrationParameterA1** (4-byte float, array size: 6):
Co-registration parameter A1. The co-registration parameters are used for calculating the position (latitude and longitude) of the observing point for each frequency except 89 GHz.

**CoRegistrationParameterA2** (4-byte float, array size: 6):
Co-registration parameter A2. The co-registration parameters are used for calculating the position (latitude and longitude) of the observing point for each frequency except 89 GHz.
continued on next figure

Figure 240: Data Format Structure for 1BASEAMSR2, AMSR2 base
Figure 241: Data Format Structure for 1BASEAMSR2, AMSR2 base
continued from last figure

- Latitude_23 4 bytes Array: npixel1 x nscan
- Longitude_23 4 bytes Array: npixel1 x nscan
- Latitude_36 4 bytes Array: npixel1 x nscan
- Longitude_36 4 bytes Array: npixel1 x nscan
- Latitude_89A 4 bytes Array: npixel2 x nscan
- Longitude_89A 4 bytes Array: npixel2 x nscan
- Latitude_89B 4 bytes Array: npixel2 x nscan
- Longitude_89B 4 bytes Array: npixel2 x nscan
- Sun_Azimuth 4 bytes Array: npixel1 x nscan
- Sun_Elevation 4 bytes Array: npixel1 x nscan
- Earth_Incidence_6 4 bytes Array: npixel1 x nscan
- Earth_Incidence_7 4 bytes Array: npixel1 x nscan
- Earth_Incidence_10 4 bytes Array: npixel1 x nscan
- Earth_Incidence_18 4 bytes Array: npixel1 x nscan
- Earth_Incidence_23 4 bytes Array: npixel1 x nscan
- Earth_Incidence_36 4 bytes Array: npixel1 x nscan
- Earth_Incidence_89A 4 bytes Array: npixel2 x nscan
- Earth_Incidence_89B 4 bytes Array: npixel2 x nscan
- Sun_Glint_Angle 4 bytes Array: npixel1 x nscan
- Earth_Azimuth 4 bytes Array: npixel1 x nscan
- Land_Ocean_Flag_6_to_36 1 byte Array: npixel1 x 6 x nscan
- Land_Ocean_Flag_89 1 byte Array: npixel2 x 2 x nscan
- Observation_Supplement 1 byte Array: 248 x nscan

continued on next figure

Figure 242: Data Format Structure for 1BASEAMSR2, AMSR2 base
continued from last figure

**1BASEAMSR2 - AMSR2 base**

- **SPC_Temperature_Count**: 2 bytes, Array: 34 x nscan
- **SPS_Temperature_Count**: 2 bytes, Array: 46 x nscan
- **PCD_Data**: 1 byte, Array: 64 x nscan
- **Scan_Data_Quality**: 1 byte, Array: 512 x nscan
- **Pixel_Data_Quality_6_to_36**: 1 byte, Array: npixel2 x nscan
- **Pixel_Data_Quality_89**: 1 byte, Array: npixel2 x nscan
- **Interpolation_Flag_6_to_36**: 1 byte, Array: count1 x freq1 x nscan
- **Interpolation_Flag_89**: 1 byte, Array: 32 x 4 x nscan
- **rfiFlag**: 2 bytes, Array: nchanRFI x npixel1 x nscan

Figure 243: Data Format Structure for 1BASEAMSR2, AMSR2 base

- **baseHeader**
  - **CoRegistrationParameterA1**: 4 bytes, Array: 6
  - **CoRegistrationParameterA2**: 4 bytes, Array: 6

Figure 244: Data Format Structure for 1BASEAMSR2, baseHeader

- **ScanTime**
  - **Year**: 2 bytes, Array: nscan
  - **Month**: 1 byte, Array: nscan
  - **DayOfMonth**: 1 byte, Array: nscan
  - **Hour**: 1 byte, Array: nscan
  - **Minute**: 1 byte, Array: nscan
  - **Second**: 1 byte, Array: nscan
  - **MilliSecond**: 2 bytes, Array: nscan
  - **DayOfYear**: 2 bytes, Array: nscan
  - **SecondOfDay**: 8 bytes, Array: nscan

Figure 245: Data Format Structure for 1BASEAMSR2, ScanTime
S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

isMissing (1-byte integer, array size: nscan):
Missing scan flag.
solarBetaAngle (4-byte float, array size: nsCan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -59.0 to 59.0 degrees. Special values are defined as: 
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nsCan):
The estimated duration in seconds since the last entry into the Earth’s shadow.

sclat (4-byte float, array size: nsCan):
Spacecraft latitude. Values range from -90 to 90.0 degree. Special values are defined as: 
-9999.9 Missing value

sclon (4-byte float, array size: nsCan):
Spacecraft longitude. Values range from -180 to 180.0 degree. Special values are defined as: 
-9999.9 Missing value

scalt (4-byte float, array size: nsCan):
Spacecraft altitude. Values range from 0 to 1000 km. Special values are defined as: 
-9999.9 Missing value

Amsr2_Scan_Time (8-byte float, array size: nsCan):
The observation start time of 89GHz A-horn. This time is a total second (TAI) from 00:00 (UTC) on January 1st, 1993.

Position_in_Orbit (8-byte float, array size: nsCan):
Fractional orbit number.

Navigation_Data (4-byte float, array size: 6 x nsCan):
Satellite position and velocity corresponding to the observation start time (Amsr2_Scan_Time) of each scan. Data is in WGS84 earth fixed coordinate system.

Attitude_Data (4-byte float, array size: 3 x nsCan):
Roll, Pitch and Yaw.

Tb_6GHz_V (4-byte float, array size: npixel1 x nsCan):
Brightness temperature.

Tb_6GHz_H (4-byte float, array size: npixel1 x nsCan):
Brightness temperature.

Tb_7GHz_V (4-byte float, array size: npixel1 x nsCan):
Brightness temperature.

Tb_7GHz_H (4-byte float, array size: npixel1 x nsCan):
Brightness temperature.

Tb_10GHz_V (4-byte float, array size: npixel1 x nsCan):
Brightness temperature.

Tb_10GHz_H (4-byte float, array size: npixel1 x nsCan):
Brightness temperature.
Tb\_18GHz\_V (4-byte float, array size: npixel1 x nscan): Brightness temperature.

Tb\_18GHz\_H (4-byte float, array size: npixel1 x nscan): Brightness temperature.

Tb\_23GHz\_V (4-byte float, array size: npixel1 x nscan): Brightness temperature.

Tb\_23GHz\_H (4-byte float, array size: npixel1 x nscan): Brightness temperature.

Tb\_36GHz\_V (4-byte float, array size: npixel1 x nscan): Brightness temperature.

Tb\_36GHz\_H (4-byte float, array size: npixel1 x nscan): Brightness temperature.

Tb\_89GHz\_V\_A (4-byte float, array size: npixel2 x nscan): Brightness temperature.

Tb\_89GHz\_H\_A (4-byte float, array size: npixel2 x nscan): Brightness temperature.

Tb\_89GHz\_V\_B (4-byte float, array size: npixel2 x nscan): Brightness temperature.

Tb\_89GHz\_H\_B (4-byte float, array size: npixel2 x nscan): Brightness temperature.

Hot\_Load\_Count\_6\_to\_36 (2-byte integer, array size: count1 x freq1 x nscan): The observed count of HTS and polarization for each frequency except 89 GHz.

Hot\_Load\_Count\_89 (2-byte integer, array size: count2 x freq2 x nscan): The observed count of HTS and polarization for 89 GHz.

Cold\_Sky\_Mirror\_Count\_6\_to\_36 (2-byte integer, array size: count1 x freq1 x nscan): The observed count of CSM and polarization for each frequency except 89 GHz.

Cold\_Sky\_Mirror\_Count\_89 (2-byte integer, array size: count2 x freq2 x nscan): The observed count of CSM and polarization for 89 GHz.

Rx\_Offset\_Gain\_Count (2-byte unsigned integer, array size: 32 x nscan): The gain and offset values for receiver (RX).

Latitude\_6 (4-byte float, array size: npixel1 x nscan): Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 6 GHz coregistration parameters to the 89A GHz location.

Longitude\_6 (4-byte float, array size: npixel1 x nscan): Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 6 GHz coregistration parameters to the 89A GHz location.
Latitude_7 (4-byte float, array size: npixel1 x nscan):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 7 GHz coregistration parameters to the 89A GHz location.

Longitude_7 (4-byte float, array size: npixel1 x nscan):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 7 GHz coregistration parameters to the 89A GHz location.

Latitude_10 (4-byte float, array size: npixel1 x nscan):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 10 GHz coregistration parameters to the 89A GHz location.

Longitude_10 (4-byte float, array size: npixel1 x nscan):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 10 GHz coregistration parameters to the 89A GHz location.

Latitude_18 (4-byte float, array size: npixel1 x nscan):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 18 GHz coregistration parameters to the 89A GHz location.

Longitude_18 (4-byte float, array size: npixel1 x nscan):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 18 GHz coregistration parameters to the 89A GHz location.

Latitude_23 (4-byte float, array size: npixel1 x nscan):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

Longitude_23 (4-byte float, array size: npixel1 x nscan):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

Latitude_36 (4-byte float, array size: npixel1 x nscan):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 36 GHz coregistration parameters to the 89A GHz location.

Longitude_36 (4-byte float, array size: npixel1 x nscan):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 36 GHz coregistration parameters to the 89A GHz location.

Latitude_89A (4-byte float, array size: npixel2 x nscan):
Latitude of the observation point on the earth surface at 89GHz A-horn.

Longitude_89A (4-byte float, array size: npixel2 x nscan):
Longitude of the observation point on the earth surface at 89GHz A-horn.

Latitude_89B (4-byte float, array size: npixel2 x nscan):
Latitude of the observation point on the earth surface at 89GHz B-horn.

Longitude_89B (4-byte float, array size: npixel2 x nscan):
Longitude of the observation point on the earth surface at 89GHz B-horn.
Sun_Azimuth (4-byte float, array size: npixel1 x nscan):
The sun azimuth angle on odd observation points (origin 1) of 89 GHz A-horn.

Sun_Elevation (4-byte float, array size: npixel1 x nscan):
The sun elevation angle on odd observation points (origin 1) of 89 GHz A-horn.

Earth_Incidence_6 (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 6 GHz.

Earth_Incidence_7 (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 7 GHz.

Earth_Incidence_10 (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 10 GHz.

Earth_Incidence_18 (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 18 GHz.

Earth_Incidence_23 (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 23 GHz.

Earth_Incidence_36 (4-byte float, array size: npixel1 x nscan):
The earth incidence angle 36 GHz.

Earth_Incidence_89A (4-byte float, array size: npixel2 x nscan):
The earth incidence angle 89 GHz A.

Earth_Incidence_89B (4-byte float, array size: npixel2 x nscan):
The earth incidence angle 89 GHz B.

Sun_Glint_Angle (4-byte float, array size: npixel1 x nscan):
Sun glint angle calculated for 89A odd numbered pixels.

Earth_Azimuth (4-byte float, array size: npixel1 x nscan):
The earth azimuth angle on odd observation points (origin 1) of 89 GHz A-horn.

Land_Ocean_Flag_6_to_36 (1-byte char, array size: npixel1 x 6 x nscan):
The land coverage percentage of the observation footprint.

Land_Ocean_Flag_89 (1-byte char, array size: npixel2 x 2 x nscan):
The land coverage percentage of the observation footprint.

Observation_Supplement (1-byte char, array size: 248 x nscan):
Observation supplement raw data such as a H/W state. If the scan is missing data, all 1 are stored in it.

SPC_Temperature_Count (2-byte unsigned integer, array size: 34 x nscan):
The temperature of SPC (Signal Processor Control unit) in each scan is stored with the value of 12 bits of raw data acquired from the satellite. If it is a missing scan, all 1 are stored in it.

SPS_Temperature_Count (2-byte unsigned integer, array size: 46 x nscan):
The temperature of SPS (Signal Processor Sensor unit) in each scan is stored with the value of 12 bits of raw data acquired from the satellite. If it is a missing scan, all 1 are stored in it.
**PCD Data** (1-byte char, array size: 64 x nscan):
The PCD (Payload Correction Data) data ID. If the scan is missing, 1 is stored in all bits.

**Scan Data Quality** (1-byte char, array size: 512 x nscan):
Array of scan data quality information and supplementary information flags. These correspond to observation data and calculation result in each scan. See AMSR2 Level 1 Product Format Specification for details.

**Pixel Data Quality 6 to 36** (1-byte char, array size: npixel2 x nscan):
Pixel quality bit flags for frequencies 6 GHz to 36 GHz. See AMSR2 Level 1 Product Format Specification for details.

**Pixel Data Quality 89** (1-byte char, array size: npixel2 x nscan):
Pixel quality bit flags for frequency 89 GHz. See AMSR2 Level 1 Product Format Specification for details.

**Interpolation Flag 6 to 36** (1-byte char, array size: count1 x freq1 x nscan):
The interpolation flag for CSM data.

**Interpolation Flag 89** (1-byte char, array size: 32 x 4 x nscan):
The interpolation flag for CSM data.

**rfiFlag** (2-byte integer, array size: nchanRFI x npixel1 x nscan):
Radio frequency Interference (RFI) Flag for channels 6V, 6H, 7V, 7H, 10V, 10H, 18V and 18H. The flag is set to 1 if the pixel is contaminated by RFI.

C Structure Header file:

```c
#ifndef _TK_1BASEAMSR2_H_
#define _TK_1BASEAMSR2_H_

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif
```
#ifndef _L1BASEAMSR2_S1_
#define _L1BASEAMSR2_S1_

typedef struct {
    SCANTIME ScanTime;
    signed char isMissing;
    float solarBetaAngle;
    float timeSinceEclipseEntry;
    float sclat;
    float slon;
    float scalt;
    double Amsr2_Scan_Time;
    double Position_in_Orbit;
    float Navigation_Data[6];
    float Attitude_Data[3];
    float Tb_6GHz_V[243];
    float Tb_6GHz_H[243];
    float Tb_7GHz_V[243];
    float Tb_7GHz_H[243];
    float Tb_10GHz_V[243];
    float Tb_10GHz_H[243];
    float Tb_18GHz_V[243];
    float Tb_18GHz_H[243];
    float Tb_23GHz_V[243];
    float Tb_23GHz_H[243];
    float Tb_36GHz_V[243];
    float Tb_36GHz_H[243];
    float Tb_89GHz_V_A[486];
    float Tb_89GHz_H_A[486];
    float Tb_89GHz_V_B[486];
    float Tb_89GHz_H_B[486];
    short Hot_Load_Count_6_to_36[12][16];
    short Hot_Load_Count_89[4][32];
    short Cold_Sky_Mirror_Count_6_to_36[12][16];
    short Cold_Sky_Mirror_Count_89[4][32];
    unsigned short Rx_Offset_Gain_Count[32];
    float Latitude_6[243];
    float Longitude_6[243];
    float Latitude_7[243];
    float Longitude_7[243];
    float Latitude_10[243];
    float Longitude_10[243];
float Latitude_18[243];
float Longitude_18[243];
float Latitude_23[243];
float Longitude_23[243];
float Latitude_36[243];
float Longitude_36[243];
float Latitude_89A[486];
float Longitude_89A[486];
float Latitude_89B[486];
float Longitude_89B[486];
float Sun_Azimuth[243];
float Sun_Elevation[243];
float Earth_Incidence_6[243];
float Earth_Incidence_7[243];
float Earth_Incidence_10[243];
float Earth_Incidence_18[243];
float Earth_Incidence_23[243];
float Earth_Incidence_36[243];
float Earth_Incidence_89A[486];
float Earth_Incidence_89B[486];
float Sun_Glint_Angle[243];
float Earth_Azimuth[243];
unsigned char Land_Ocean_Flag_6_to_36[6][243];
unsigned char Land_Ocean_Flag_89[2][486];
unsigned char Observation_Supplement[248];
unsigned short SPC_Temperature_Count[34];
unsigned short SPS_Temperature_Count[46];
unsigned char PCD_Data[64];
unsigned char Scan_Data_Quality[512];
unsigned char Pixel_Data_Quality_6_to_36[486];
unsigned char Pixel_Data_Quality_89[486];
unsigned char Interpolation_Flag_6_to_36[12][16];
unsigned char Interpolation_Flag_89[4][32];
short rfiFlag[243][8];
} L1BASEAMSR2_S1;

#endif

#ifndef _L1BASEAMSR2_BASEHEADER_
define _L1BASEAMSR2_BASEHEADER_

typedef struct {
    float CoRegistrationParameterA1[6];

#endif

#ifndef _L1BASEAMSR2_BASEHEADER_
float CoRegistrationParameterA2[6];
} L1BASEAMSR2_BASEHEADER;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASEAMSR2_S1/
  RECORD /SCANTIME/ ScanTime
  BYTE isMissing
  REAL*4 solarBetaAngle
  REAL*4 timeSinceEclipseEntry
  REAL*4 sclat
  REAL*4 sclon
  REAL*4 scalt
  REAL*8 Amsr2_Scan_Time
  REAL*8 Position_in_Orbit
  REAL*4 Navigation_Data(6)
  REAL*4 Attitude_Data(3)
  REAL*4 Tb_6GHz_V(243)
  REAL*4 Tb_6GHz_H(243)
  REAL*4 Tb_7GHz_V(243)
  REAL*4 Tb_7GHz_H(243)
  REAL*4 Tb_10GHz_V(243)
  REAL*4 Tb_10GHz_H(243)
  REAL*4 Tb_18GHz_V(243)
  REAL*4 Tb_18GHz_H(243)
  REAL*4 Tb_23GHz_V(243)
  REAL*4 Tb_23GHz_H(243)
REAL*4 Tb_36GHz_V(243)
REAL*4 Tb_36GHz_H(243)
REAL*4 Tb_89GHz_V_A(486)
REAL*4 Tb_89GHz_H_A(486)
REAL*4 Tb_89GHz_V_B(486)
REAL*4 Tb_89GHz_H_B(486)
INTEGER*2 Hot_Load_Count_6_to_36(16,12)
INTEGER*2 Hot_Load_Count_89(32,4)
INTEGER*2 Cold_Sky_Mirror_Count_6_to_36(16,12)
INTEGER*2 Cold_Sky_Mirror_Count_89(32,4)
INTEGER*2 Rx_Offset_Gain_Count(32)
REAL*4 Latitude_6(243)
REAL*4 Longitude_6(243)
REAL*4 Latitude_7(243)
REAL*4 Longitude_7(243)
REAL*4 Latitude_10(243)
REAL*4 Longitude_10(243)
REAL*4 Latitude_18(243)
REAL*4 Longitude_18(243)
REAL*4 Latitude_23(243)
REAL*4 Longitude_23(243)
REAL*4 Latitude_36(243)
REAL*4 Longitude_36(243)
REAL*4 Latitude_89A(486)
REAL*4 Longitude_89A(486)
REAL*4 Latitude_89B(486)
REAL*4 Longitude_89B(486)
REAL*4 Sun_Azimuth(243)
REAL*4 Sun_Elevation(243)
REAL*4 Earth_Incidence_6(243)
REAL*4 Earth_Incidence_7(243)
REAL*4 Earth_Incidence_10(243)
REAL*4 Earth_Incidence_18(243)
REAL*4 Earth_Incidence_23(243)
REAL*4 Earth_Incidence_36(243)
REAL*4 Earth_Incidence_89A(486)
REAL*4 Earth_Incidence_89B(486)
REAL*4 Sun_Glint_Angle(243)
REAL*4 Earth_Azimuth(243)
CHARACTER Land_Ocean_Flag_6_to_36(243,6)
CHARACTER Land_Ocean_Flag_89(486,2)
CHARACTER Observation_Supplement(248)
INTEGER*2 SPC_Temperature_Count(34)
INTEGER*2 SPS_Temperature_Count(46)
CHARACTER PCD_Data(64)
CHARACTER Scan_Data_Quality(512)
CHARACTER Pixel_Data_Quality_6_to_36(486)
CHARACTER Pixel_Data_Quality_89(486)
CHARACTER Interpolation_Flag_6_to_36(16,12)
CHARACTER Interpolation_Flag_89(32,4)
INTEGER*2 rfiFlag(8,243)
END STRUCTURE

STRUCTURE /L1BASEAMSR2_BASEHEADER/
  REAL*4 CoRegistrationParameterA1(6)
  REAL*4 CoRegistrationParameterA2(6)
END STRUCTURE

5.12 1BASEWIND - Windsat base

1BASEWIND contains brightness temperature from the WindSat passive microwave instrument flown on the Coriolis satellite. Swath S1 is the only swath. All of the data of the WindSat Sensor Data Record (SDR) data files are included. See the SDR documentation for details.

Dimension definitions:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nscan1</td>
<td>var Number of Swath 1 scans in the granule.</td>
</tr>
<tr>
<td>nchannel1</td>
<td>16 Number of Swath 1 channels.</td>
</tr>
<tr>
<td>npixel1</td>
<td>80 Number of Swath 1 pixels in one scan.</td>
</tr>
<tr>
<td>nchUIA1</td>
<td>5 Number of Swath S1 unique incidence angles.</td>
</tr>
<tr>
<td>nspdim</td>
<td>3 Number of spacial dimensions.</td>
</tr>
<tr>
<td>nerr</td>
<td>27 Number of error flags.</td>
</tr>
<tr>
<td>nhorn</td>
<td>6 Number of horns.</td>
</tr>
<tr>
<td>nresamplingFlag</td>
<td>22 Number of resampling flags.</td>
</tr>
<tr>
<td>nrfiFlag</td>
<td>5 Number of RFI flags.</td>
</tr>
</tbody>
</table>

Figure 246 through Figure 249 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the
Figure 246: Data Format Structure for 1BASEWIND, Windsat base
continued from last figure

S1

- rSatECF 4 bytes Array: nspdim x npixel1 x nscan1
- einDiffs 4 bytes Array: nhorn x npixel1 x nscan1
- scan 4 bytes Array: npixel1 x nscan1
- land2water 1 byte Array: npixel1 x nscan1
- water2land 1 byte Array: npixel1 x nscan1
- surfaceType 1 byte Array: npixel1 x nscan1
- errorFlag 1 byte Array: nerr x npixel1 x nscan1
- downCount 4 bytes Array: npixel1 x nscan1
- sunGlintAngle 4 bytes Array: npixel1 x nscan1
- resamplingFlags 1 byte Array: nresamplingFlag x npixel1 x nscan1
- praDiffs 4 bytes Array: nspdim x npixel1 x nscan1
- rfiFlags 1 byte Array: nrfiFlag x npixel1 x nscan1

Figure 247: Data Format Structure for 1BASEWIND, Windsat base

baseHeader

- TLE.date 4 bytes Array: 6
- TLE.time 8 bytes Array: 1

Figure 248: Data Format Structure for 1BASEWIND, baseHeader

ScanTime

- Year 2 bytes Array: nscan1
- Month 1 byte Array: nscan1
- DayOfMonth 1 byte Array: nscan1
- Hour 1 byte Array: nscan1
- Minute 1 byte Array: nscan1
- Second 1 byte Array: nscan1
- MilliSecond 2 bytes Array: nscan1
- DayOfYear 2 bytes Array: nscan1
- SecondOfDay 8 bytes Array: nscan1

Figure 249: Data Format Structure for 1BASEWIND, ScanTime
same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**baseHeader** (Group)

**TLE_date** (4-byte integer, array size: 6):
TLE date time arrays.

**TLE_time** (8-byte float, array size: 1):
TLE time as in two line element.

**S1** (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value
Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime.sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

isMissing (1-byte integer, array size: nscan1):
Missing scan flag.

solarBetaAngle (4-byte float, array size: nscan1):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9  Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan1):
The estimated duration in seconds since the last entry into the Earth’s shadow.

sclat (4-byte float, array size: nscan1):
Spacecraft latitude. Values range from -90 to 90.0 degrees. Special values are defined as:
-9999.9  Missing value

sclon (4-byte float, array size: nscan1):
Spacecraft longitude. Values range from -180 to 180.0 degrees. Special values are defined as:
-9999.9  Missing value

scalt (4-byte float, array size: nscan1):
Spacecraft altitude. Values range from 0 to 1000 km. Special values are defined as:
-9999.9  Missing value

JD2000 (8-byte float, array size: npixel1 x nscan1):
Time of the measurement, seconds since noon January 1, 2000.
tbs (4-byte float, array size: nchannel1 x npixel1 x nscan1): Brightness Temperatures for 6.8 GHz (V,H), 10.7 GHz (V,H,U,F), 18.7 GHz (V,H,U,F), 23.8GHz (V,H) and 37 GHz (V,H,U,F).

scanAngle (4-byte float, array size: npixel1 x nscan1): The angle between the flight and look directions.

Latitude (4-byte float, array size: npixel1 x nscan1): The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel1 x nscan1): The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

EIA (4-byte float, array size: nchUIA1 x npixel1 x nscan1): Earth incidence angles for 6.8, 10.7, 18.7, 23.8 and 37 GHz.

PRA (4-byte float, array size: nchUIA1 x npixel1 x nscan1): Polar rotation angles for 6.8, 10.7, 18.7, 23.8 and 37 GHz.

CAA (4-byte float, array size: npixel1 x nscan1): Compass azimuth angle at the pierce point. North equals 0 degree, and the CAA increases clockwise.


rlos_ned (4-byte float, array size: nspdim x npixel1 x nscan1): Line of sight vector in north-east-down coordinates (in meters). Not sure about the range.

rSatECF (4-byte float, array size: nspdim x npixel1 x nscan1): Location of the satellite at the 37 GHz VH measurement in Earth Centered Fixed (ECF) coordinate system. Not sure about the range.

eiaDiffs (4-byte float, array size: nhorn x npixel1 x nscan1): EIA difference between horns in degrees. The order is 10GHz PM-VH, 10GHz LR-VH, 18GHz PM-VH, 18GHz LR-VH, 37GHz PM-VH, 37GHz LR-VH.

scan (4-byte integer, array size: npixel1 x nscan1): Scan Number.

land2water (1-byte integer, array size: npixel1 x nscan1): Amount of land contamination in a water pixel expressed in parts per thousand (PPT), 127 for greater than 100 PPT.
**water2land** (1-byte integer, array size: npixel1 x nscan1):
Amount of water contamination in a land pixel expressed in parts per thousand (PPT),
127 for greater than 100 PPT.

**surfaceType** (1-byte integer, array size: npixel1 x nscan1):
Surface type.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Land</td>
</tr>
<tr>
<td>1</td>
<td>Not Used</td>
</tr>
<tr>
<td>2</td>
<td>Near Coast</td>
</tr>
<tr>
<td>3</td>
<td>Ice</td>
</tr>
<tr>
<td>4</td>
<td>Possible Ice</td>
</tr>
<tr>
<td>5</td>
<td>Ocean</td>
</tr>
<tr>
<td>6</td>
<td>Coast</td>
</tr>
<tr>
<td>7</td>
<td>Spare</td>
</tr>
</tbody>
</table>

**errorFlag** (1-byte integer, array size: nerr x npixel1 x nscan1): There are 27 Error flags unpacked from the SDR file.

1. Gain saturation flag for 6.8 GHz.
   Set to 1 when strong RFI causes the gain to change.
2. Gain saturation flag for 10.7 GHz.
3. Gain saturation flag for 18.7 GHz.
4. Gain saturation flag for 23.8 GHz.
5. Gain saturation flag for 37.0 GHz.
6. Not used.
7. Not used.
8. Not used.
9. Forward/aft flag. 1 for forward, 0 for aft.
10. Ascending/descending flag. 1 for ascending, 0 for descending.
11. Warning for probable solar disturbance of warm load if value is 1.
12. Corrected warm load gains applied if value is 1.
13. Sun glare angle invalid because no sun vector or LOS does not pierce earth if value is 1.
14. Sun glare angle. 0–30 for 0–60 degrees, 31 is greater than 60 degrees.
15. Cold load flag for 6.8 GHz. Cold sky calibration data was contaminated with RFI or lunar intrusion if value is 1.
16. Cold load flag for 10.7 GHz.
17. Cold load flag for 18.7 GHz.
18. Cold load flag for 23.8 GHz.
19. Cold load flag for 37.0 GHz.
20. Warm load flag for 6.8 GHz. This flag indicates presence of thermal gradients on the warm load if value is 1.
21. Warm load flag for 10.7 GHz.
22. Warm load flag for 18.7 GHz.
23. Warm load flag for 23.8 GHz.
24. Warm load flag for 37.0 GHz.
25. Satellite attitude transient if value is 1.
26. Star viewer outage near attitude transient if value is 1.
27. Not used.

downCount (4-byte integer, array size: npixel1 x nscan1):
Pixel Number along scan. Not sure about the range.

sunGlintAngle (4-byte float, array size: npixel1 x nscan1):
Sun glint angle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection.

resamplingFlags (1-byte integer, array size: nresamplingFlag x npixel1 x nscan1):
Resampling flags are set to 1 if the resampling percentage threshold was not met. There are 22 resampling flags: 6VH 10VHPMLR 18VHPMLR 23VH 37VHPMLR.

praDiffs (4-byte float, array size: nspdim x npixel1 x nscan1):
PM PRA minus VH PRA, in degrees. The order is 10GHz PM-VH, 18GHz PM-VH, 37GHz PM-VH.

rfiFlags (1-byte integer, array size: nrfiFlag x npixel1 x nscan1):
Flags for ocean-reflected RFI. There are 5 RFI flags (6.8, 10.7, 18.7, 23.8 and 37 GHz), only first 3 are implemented.

C Structure Header file:
typedef struct {
    SCANTIME ScanTime;
    signed char isMissing;
    float solarBetaAngle;
    float timeSinceEclipseEntry;
    float sclat;
    float sclon;
    float scalt;
    double JD2000[80];
    float tbs[80][16];
    float scanAngle[80];
    float Latitude[80];
    float Longitude[80];
    float EIA[80][5];
    float PRA[80][5];
    float CAA[80];
    float tbDiffs[80][6];
    float rlos_ned[80][3];
    float rSatECF[80][3];
    float eiaDiffs[80][6];
    int scan[80];
    signed char land2water[80];
    signed char water2land[80];
    signed char surfaceType[80];
    signed char errorFlag[80][27];
    int downCount[80];
    float sunGlintAngle[80];
    signed char resamplingFlags[80][22];
    float pradiffs[80][3];
    signed char rfiFlags[80][5];
} L1BASEWIND_S1;
typedef struct {
    int TLE_date[6];
    double TLE_time[1];
} L1BASEWIND_BASEHEADER;

#endif

#endif

**Fortran Structure Header file:**

**STRUCTURE /SCANTIME/**

- INTEGER*2 Year
- BYTE Month
- BYTE DayOfMonth
- BYTE Hour
- BYTE Minute
- BYTE Second
- INTEGER*2 MilliSecond
- INTEGER*2 DayOfYear
- REAL*8 SecondOfDay

**END STRUCTURE**

**STRUCTURE /L1BASEWIND_S1/**

- RECORD /SCANTIME/ ScanTime
- BYTE isMissing
- REAL*4 solarBetaAngle
- REAL*4 timeSinceEclipseEntry
- REAL*4 sclat
- REAL*4 sclon
- REAL*4 scalt
- REAL*8 JD2000(80)
- REAL*4 tbs(16,80)
- REAL*4 scanAngle(80)
- REAL*4 Latitude(80)
- REAL*4 Longitude(80)
- REAL*4 EIA(5,80)
- REAL*4 PRA(5,80)
- REAL*4 CAA(80)
- REAL*4 tbDiffs(6,80)
- REAL*4 rlos_ned(3,80)
- REAL*4 rSatECF(3,80)
REAL*4 eiaDiffs(6,80)
INTEGER*4 scan(80)
BYTE land2water(80)
BYTE water2land(80)
BYTE surfaceType(80)
BYTE errorFlag(27,80)
INTEGER*4 downCount(80)
REAL*4 sunGlintAngle(80)
BYTE resamplingFlags(22,80)
REAL*4 praDiffs(3,80)
BYTE rfiFlags(5,80)
END STRUCTURE

STRUCTURE /L1BASEWIND_BASEHEADER/
  INTEGER*4 TLE_date(6)
  REAL*8 TLE_time(1)
END STRUCTURE

5.13 1BASEAMSUA - AMSUA base

1BASEAMSUA contains antenna temperature from the AMSUA passive microwave instrument flown on the NOAA and METOPS satellites. Swath S1 is the only swath.

Dimension definitions:
  nscan1   var Number of Swath 1 scans in the granule.
  nchannel1 15 Number of Swath 1 channels.
  npixel1 30 Number of Swath 1 pixels in one scan.
  nchUIA1 1 Number of Swath S1 unique incidence angles.
  three  3 Number of spacial dimensions.

Figure 250 through Figure 253 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in
5.13 1BASEAMSUA - AMSUA base

Figure 250: Data Format Structure for 1BASEAMSUA, AMSUA base

continued on next figure
continued from last figure

---

**Figure 251: Data Format Structure for 1BASEAMSUA, AMSUA base**

- Chan1AT: 4 bytes, Array: npixel1 x nscan1
- Chan2AT: 4 bytes, Array: npixel1 x nscan1
- Chan3AT: 4 bytes, Array: npixel1 x nscan1
- Chan4AT: 4 bytes, Array: npixel1 x nscan1
- Chan5AT: 4 bytes, Array: npixel1 x nscan1
- Chan6AT: 4 bytes, Array: npixel1 x nscan1
- Chan7AT: 4 bytes, Array: npixel1 x nscan1
- Chan8AT: 4 bytes, Array: npixel1 x nscan1
- Chan9AT: 4 bytes, Array: npixel1 x nscan1
- Chan10AT: 4 bytes, Array: npixel1 x nscan1
- Chan11AT: 4 bytes, Array: npixel1 x nscan1
- Chan12AT: 4 bytes, Array: npixel1 x nscan1
- Chan13AT: 4 bytes, Array: npixel1 x nscan1
- Chan14AT: 4 bytes, Array: npixel1 x nscan1
- Chan15AT: 4 bytes, Array: npixel1 x nscan1
- TPW: 4 bytes, Array: npixel1 x nscan1
- CLW: 4 bytes, Array: npixel1 x nscan1
- Sice: 4 bytes, Array: npixel1 x nscan1
- Tsfc: 4 bytes, Array: npixel1 x nscan1
- Emis23: 4 bytes, Array: npixel1 x nscan1
- Emis31: 4 bytes, Array: npixel1 x nscan1
- Emis50: 4 bytes, Array: npixel1 x nscan1

---

**Figure 252: Data Format Structure for 1BASEAMSUA, baseHeader**

- TLE_date: 4 bytes, Array: 6
- TLE_time: 8 bytes, Array: 1

---
5.13 1BASEAMSUA - AMSUA base

Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

baseHeader (Group)

TLE_date (4-byte integer, array size: 6):
TLE date time arrays.

TLE_time (8-byte float, array size: 1):
TLE time as in two line element.

S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.
Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
isMissing (1-byte integer, array size: nscan1):
Missing scan flag.

sunGlintAngle (4-byte float, array size: npixel1 x nscan1):
Unpacked sun glint angle. Not sure about the range.

solarBetaAngle (4-byte float, array size: nscan1):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given
by the cross product of the spacecraft position and velocity vectors. Values range from
-59.0 to 59.0 degrees. Special values are defined as:
-9999.9   Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan1):
The estimated duration in seconds since the last entry into the Earth’s shadow.

tlePos (4-byte float, array size: three x nscan1):
TLE satellite position. Values are in m. Special values are defined as:
-9999.9   Missing value

tleVel (4-byte float, array size: three x nscan1):
TLE satellite velocity. Values are in m/s. Special values are defined as:
-9999.9   Missing value

sclat (4-byte float, array size: nscan1):
Spacecraft latitude. Values range from -90 to 90.0 degree. Special values are defined as:
-9999.9   Missing value

sclon (4-byte float, array size: nscan1):
Spacecraft longitude. Values range from -180 to 180.0 degree. Special values are defined
as:
-9999.9   Missing value

scalt (4-byte float, array size: nscan1):
Spacecraft altitude. Values range from 0 to 1000 km. Special values are defined as:
-9999.9   Missing value

TimeTAI93 (8-byte float, array size: nscan1):
Number of seconds since 0000 Jan 1, 1993.

sfcType (1-byte integer, array size: npixel1 x nscan1):
Surface type: 0=ocean, 1=land, 2=coast

orbitMode (1-byte integer, array size: nscan1):
Orbit direction: 1=ascending 2=descending.

LZangle (4-byte float, array size: npixel1 x nscan1):
Local zenith angle. Not sure about the range and units.

SZangle (4-byte float, array size: npixel1 x nscan1):
Solar zenith angle. Not sure about the range and units.

Chan1AT (4-byte float, array size: npixel1 x nscan1):
Channel 1 Antenna Temperature.
Chan2AT (4-byte float, array size: npixel1 x nscan1): Channel 2 Antenna Temperature.
Chan3AT (4-byte float, array size: npixel1 x nscan1): Channel 3 Antenna Temperature.
Chan4AT (4-byte float, array size: npixel1 x nscan1): Channel 4 Antenna Temperature.
Chan5AT (4-byte float, array size: npixel1 x nscan1): Channel 5 Antenna Temperature.
Chan6AT (4-byte float, array size: npixel1 x nscan1): Channel 6 Antenna Temperature.
Chan7AT (4-byte float, array size: npixel1 x nscan1): Channel 7 Antenna Temperature.
Chan8AT (4-byte float, array size: npixel1 x nscan1): Channel 8 Antenna Temperature.
Chan9AT (4-byte float, array size: npixel1 x nscan1): Channel 9 Antenna Temperature.
Chan10AT (4-byte float, array size: npixel1 x nscan1): Channel 10 Antenna Temperature.
Chan11AT (4-byte float, array size: npixel1 x nscan1): Channel 11 Antenna Temperature.
Chan12AT (4-byte float, array size: npixel1 x nscan1): Channel 12 Antenna Temperature.
Chan13AT (4-byte float, array size: npixel1 x nscan1): Channel 13 Antenna Temperature.
Chan14AT (4-byte float, array size: npixel1 x nscan1): Channel 14 Antenna Temperature.
Chan15AT (4-byte float, array size: npixel1 x nscan1): Channel 15 Antenna Temperature.
TPW (4-byte float, array size: npixel1 x nscan1): Total Precipitable Water.
CLW (4-byte float, array size: npixel1 x nscan1): Cloud Liquid Water.
Sice (4-byte float, array size: npixel1 x nscan1): Sea Ice Concentration.
Tsfc (4-byte float, array size: npixel1 x nscan1): Surface Temperature.
Emis23 (4-byte float, array size: npixel1 x nscan1): Emissivity at 23.8 GHz.
Emis31 (4-byte float, array size: npixel1 x nscan1): Emissivity at 31.4 GHz.

Emis50 (4-byte float, array size: npixel1 x nscan1): Emissivity at 50.3 GHz.

C Structure Header file:

```c
#ifndef _TK_1BASEAMSUA_H_
#define _TK_1BASEAMSUA_H_

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#endif

#ifndef _L1BASEAMSUA_S1_
#define _L1BASEAMSUA_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[30];
    float Longitude[30];
    signed char isMissing;
    float sunGlintAngle[30];
    float solarBetaAngle;
    float timeSinceEclipseEntry;
    float tlePos[3];
    float tleVel[3];
    float sclat;
    float sclon;
    float scalt;
} L1BASEAMSUA_S1;
#endif
```
double TimeTAI93;
signed char sfcType[30];
signed char orbitMode;
float LZangle[30];
float SZangle[30];
float Chan1AT[30];
float Chan2AT[30];
float Chan3AT[30];
float Chan4AT[30];
float Chan5AT[30];
float Chan6AT[30];
float Chan7AT[30];
float Chan8AT[30];
float Chan9AT[30];
float Chan10AT[30];
float Chan11AT[30];
float Chan12AT[30];
float Chan13AT[30];
float Chan14AT[30];
float Chan15AT[30];
float TPW[30];
float CLW[30];
float Sice[30];
float Tsfc[30];
float Emis23[30];
float Emis31[30];
float Emis50[30];
} L1BASEAMSUA_S1;
#endif

#include "L1BASEAMSUA_BASEHEADER_.h"

#define _L1BASEAMSUA_BASEHEADER_

typedef struct {
    int TLE_date[6];
    double TLE_time[1];
} L1BASEAMSUA_BASEHEADER;

#endif

Fortran Structure Header file:
STRUCTURE /SCANTIME/
   INTEGER*2 Year
   BYTE Month
   BYTE DayOfMonth
   BYTE Hour
   BYTE Minute
   BYTE Second
   INTEGER*2 MilliSecond
   INTEGER*2 DayOfYear
   REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASEAMSUA_S1/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(30)
   REAL*4 Longitude(30)
   BYTE isMissing
   REAL*4 sunGlintAngle(30)
   REAL*4 solarBetaAngle
   REAL*4 timeSinceEclipseEntry
   REAL*4 tlePos(3)
   REAL*4 tleVel(3)
   REAL*4 sclat
   REAL*4 sclon
   REAL*4 scalt
   REAL*8 TimeTAI93
   BYTE sfcType(30)
   BYTE orbitMode
   REAL*4 LZangle(30)
   REAL*4 SZangle(30)
   REAL*4 Chan1AT(30)
   REAL*4 Chan2AT(30)
   REAL*4 Chan3AT(30)
   REAL*4 Chan4AT(30)
   REAL*4 Chan5AT(30)
   REAL*4 Chan6AT(30)
   REAL*4 Chan7AT(30)
   REAL*4 Chan8AT(30)
   REAL*4 Chan9AT(30)
   REAL*4 Chan10AT(30)
   REAL*4 Chan11AT(30)
   REAL*4 Chan12AT(30)
REAL*4 Chan13AT(30)
REAL*4 Chan14AT(30)
REAL*4 Chan15AT(30)
REAL*4 TPW(30)
REAL*4 CLW(30)
REAL*4 Sice(30)
REAL*4 Tsfc(30)
REAL*4 Emis23(30)
REAL*4 Emis31(30)
REAL*4 Emis50(30)

END STRUCTURE

STRUCTURE /L1BASEAMSUA_BASEHEADER/
  INTEGER*4 TLE_date(6)
  REAL*8 TLE_time(1)
END STRUCTURE

5.14 1BASEAMSUB - AMSUB base

1BASEAMSUB contains brightness temperature from the AMSU-B passive microwave instrument flown on the NOAA satellites. Swath S1 is the only swath. Swath S1 contains 5 channels: 89.0 +/- 0.9 GHz, 150.0 +/- 0.9 GHz, 183.31 +/- 1 GHz, 183.31 +/- 3 GHz, 183.31 +/- 7 GHz. The scan period is 2.667s. The input is Level-2 AMSU-B Orbital products in HDF-EOS format archived at CLASS. All of the data of the input are included. Brightness temperature was obtained by applying the Antena Pattern Correction to the antenna temperature. Please see the Microwave Surface and Precipitation Products System (MSPPS) User’s Manual and NOAA KLM User’s Guide for details.

Dimension definitions:
- nsclan1 var Number of Swath 1 scans in the granule.
- nchannel1 5 Number of Swath 1 channels.
- n270 270 Number of 270.
- npixel1 90 Number of Swath 1 pixels in one scan.
- nchUIA1 1 Number of Swath S1 unique incidence angles.
- three 3 Number of spacial dimensions.

Figure 254 through Figure 257 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.
Figure 254: Data Format Structure for 1BASEAMSUB, AMSUB base
continued from last figure

Figure 255: Data Format Structure for 1BASEAMSUB, AMSUB base

Figure 256: Data Format Structure for 1BASEAMSUB, baseHeader

Figure 257: Data Format Structure for 1BASEAMSUB, ScanTime
**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**baseHeader** (Group)

**TLE_date** (4-byte integer, array size: 6):
TLE date time arrays.

**TLE_time** (8-byte float, array size: 1):
TLE time as in two line element.

**accf** (8-byte float, array size: nchannel1 x n270):
The correction due to angular dependence of reflectivity.

**Bspace** (8-byte float, array size: nchannel1):
The channel offset derived from deep space calibration.

**waveNumber** (4-byte float, array size: nchannel1):
Wave number.

**S1** (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined
as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**isMissing** (1-byte integer, array size: nscan1):
Missing scan flag.
sunGlintAngle (4-byte float, array size: npixel1 x nscan1):
sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection.

solarBetaAngle (4-byte float, array size: nscan1):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -59.0 to 59.0 degrees. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan1):
The estimated duration in seconds since the last entry into the Earth’s shadow.

tlePos (4-byte float, array size: three x nscan1):
Spacecraft position at the ScanTime. Values are in m. Special values are defined as:
-9999.9 Missing value

tleVel (4-byte float, array size: three x nscan1):
Spacecraft velocity at the ScanTime. Values are in m/s. Special values are defined as:
-9999.9 Missing value

sclat (4-byte float, array size: nscan1):
Spacecraft latitude at the ScanTime. Values range from -90 to 90.0 degree. Special values are defined as:
-9999.9 Missing value

sclon (4-byte float, array size: nscan1):
Spacecraft longitude at the ScanTime. Values range from -180 to 180.0 degree. Special values are defined as:
-9999.9 Missing value

scalt (4-byte float, array size: nscan1):
Spacecraft altitude at the ScanTime. Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

TimeTAI93 (8-byte float, array size: nscan1):
Number of seconds since 0000 Jan 1, 1993.

sfcType (1-byte integer, array size: npixel1 x nscan1):
Surface type: 0=ocean, 1=land, 2=coast

orbitMode (1-byte integer, array size: nscan1):
Orbit direction: 1=ascending 2=descending.

LZangle (4-byte float, array size: npixel1 x nscan1):
Local zenith angle. Values range from -60 to 60 degrees. Special values are defined as:
-9999.9 Missing value

SZangle (4-byte float, array size: npixel1 x nscan1):
Solar zenith angle. Not sure about the range and units.
\textbf{Chan1BT} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Channel 1 brightness Temperature.

\textbf{Chan2BT} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Channel 2 brightness Temperature.

\textbf{Chan3BT} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Channel 3 brightness Temperature.

\textbf{Chan4BT} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Channel 4 brightness Temperature.

\textbf{Chan5BT} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Channel 5 brightness Temperature.

\textbf{RR} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Rain rate.

\textbf{Snow} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Snow cover.

\textbf{IWP} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Ice water path.

\textbf{SWE} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Snow water equivalent.

\textbf{SFR} (4-byte float, array size: \textit{npixel1 x nscan1}): 
Snow fall rate.

\textbf{C Structure Header file:}

\begin{verbatim}
#ifndef _TK_1BASEAMSUB_H_
define _TK_1BASEAMSUB_H_

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
\end{verbatim}
typedef struct {
    SCANTIME ScanTime;
    float Latitude[90];
    float Longitude[90];
    signed char isMissing;
    float sunGlintAngle[90];
    float solarBetaAngle;
    float timeSinceEclipseEntry;
    float tlePos[3];
    float tleVel[3];
    float sclat;
    float sclon;
    float scalt;
    double TimeTAI93;
    signed char sfcType[90];
    signed char orbitMode;
    float LZangle[90];
    float SZangle[90];
    float Chan1BT[90];
    float Chan2BT[90];
    float Chan3BT[90];
    float Chan4BT[90];
    float Chan5BT[90];
    float RR[90];
    float Snow[90];
    float IWP[90];
    float SWE[90];
    float SFR[90];
} L1BASEAMSUB_S1;

#endif

#ifndef _L1BASEAMSUB_BASEHEADER_
#define _L1BASEAMSUB_BASEHEADER_

typedef struct {
    int TLE_date[6];
    double TLE_time[1];
} L1BASEAMSUB_BASEHEADER;

#endif
double accf[270][5];
double Bspace[5];
float waveNumber[5];
} L1BASEAMSUB_BASEHEADER;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /SCANTIME/
   INTEGER*2 Year
   BYTE Month
   BYTE DayOfMonth
   BYTE Hour
   BYTE Minute
   BYTE Second
   INTEGER*2 MilliSecond
   INTEGER*2 DayOfYear
   REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASEAMSUB_S1/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(90)
   REAL*4 Longitude(90)
   BYTE isMissing
   REAL*4 sunGlintAngle(90)
   REAL*4 solarBetaAngle
   REAL*4 timeSinceEclipseEntry
   REAL*4 tlePos(3)
   REAL*4 tleVel(3)
   REAL*4 sclat
   REAL*4 sclon
   REAL*4 scalt
   REAL*8 TimeTAI93
   BYTE sfcType(90)
   BYTE orbitMode
   REAL*4 LZangle(90)
   REAL*4 SZangle(90)
   REAL*4 Chan1BT(90)
   REAL*4 Chan2BT(90)
5.15 1BASEMHS - MHS base

1BASEMHS contains brightness temperature from the MHS passive microwave instrument flown on the NOAA and METOPS satellites. Swath S1 is the only swath. Swath S1 contains 5 channels: 89.0 GHzV, 157.0 GHzV, 183.3 +/- 1 GHzH, 183.3 +/- 3 GHzH, 190.3 GHzV. MHS is very similar to AMSU-B. The scan period is 2.667s. The granule size is one half orbit. The input is Level-2 MHS Orbital products in HDF-EOS format archived at CLASS. All of the data of the input are included. Brightness temperature was obtained by applying the Antena Pattern Correction to the antena temperature. See the Microwave Surface and Precipitation Products System (MSPPS) Users’ Manual, which discussed AMSU-B. AMSU-B is the same format as MHS.

Dimension definitions:
- nscan1 var Number of Swath 1 scans in the granule.
- nchannel1 5 Number of Swath 1 channels.
- n270 270 Number of 270.
- npixel1 90 Number of Swath 1 pixels in one scan.
- nchUIA1 1 Number of Swath S1 unique incidence angles.
- three 3 Number of spacial dimensions.

Figure 258 through Figure 261 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products.
Figure 258: Data Format Structure for 1BASEMHS, MHS base
Figure 259: Data Format Structure for 1BASEMHS, MHS base

Figure 260: Data Format Structure for 1BASEMHS, baseHeader

Figure 261: Data Format Structure for 1BASEMHS, ScanTime
See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**baseHeader** (Group)

**TLE_date** (4-byte integer, array size: 6):
TLE date time arrays.

**TLE_time** (8-byte float, array size: 1):
TLE time as in two line element.

**accf** (8-byte float, array size: \text{nchannel1 x n270}):
The correction due to angular dependence of reflectivity.

**Bspace** (8-byte float, array size: \text{nchannel1}):
The channel offset derived from deep space calibration.

**waveNumber** (4-byte float, array size: \text{nchannel1}):
Wave number.

**S1** (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.
Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
isMissing (1-byte integer, array size: nscan1):
Missing scan flag.

sunGlintAngle (4-byte float, array size: npixel1 x nscan1):
sunGlintAngle is the angular separation between the Reflected Satellite View Vector and
the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the
specular (mirror-like) sun reflection.

solarBetaAngle (4-byte float, array size: nscan1):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given
by the cross product of the spacecraft position and velocity vectors. Values range from
-59.0 to 59.0 degrees. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan1):
The estimated duration in seconds since the last entry into the Earth’s shadow.

tlePos (4-byte float, array size: three x nscan1):
Spacecraft position at the ScanTime. Values are in m. Special values are defined as:
-9999.9 Missing value

tleVel (4-byte float, array size: three x nscan1):
Spacecraft velocity at the ScanTime. Values are in m/s. Special values are defined as:
-9999.9 Missing value

sclat (4-byte float, array size: nscan1):
Spacecraft latitude at the ScanTime. Values range from -90 to 90.0 degree. Special values
are defined as:
-9999.9 Missing value

sclon (4-byte float, array size: nscan1):
Spacecraft longitude at the ScanTime. Values range from -180 to 180.0 degree. Special
values are defined as:
-9999.9 Missing value

scalt (4-byte float, array size: nscan1):
Spacecraft altitude at the ScanTime. Values range from 0 to 1000 km. Special values are
defined as:
-9999.9 Missing value

TimeTAI93 (8-byte float, array size: nscan1):
Number of seconds since 0000 Jan 1, 1993.

sftype (1-byte integer, array size: npixel1 x nscan1):
Surface type: 0=oceand, 1=land, 2=coast

orbitMode (1-byte integer, array size: nscan1):
Orbit direction: 1=ascending 2=descending.

LZangle (4-byte float, array size: npixel1 x nscan1):
Local zenith angle. Values range from -60 to 60 degrees. Special values are defined as:
-9999.9 Missing value
**SZangle** (4-byte float, array size: npixel1 x nscan1):
Solar zenith angle. Not sure about the range and units.

**Chan1BT** (4-byte float, array size: npixel1 x nscan1):
Channel 1 brightness Temperature.

**Chan2BT** (4-byte float, array size: npixel1 x nscan1):
Channel 2 brightness Temperature.

**Chan3BT** (4-byte float, array size: npixel1 x nscan1):
Channel 3 brightness Temperature.

**Chan4BT** (4-byte float, array size: npixel1 x nscan1):
Channel 4 brightness Temperature.

**Chan5BT** (4-byte float, array size: npixel1 x nscan1):
Channel 5 brightness Temperature.

**RR** (4-byte float, array size: npixel1 x nscan1):
Rain rate.

**Snow** (4-byte float, array size: npixel1 x nscan1):
Snow cover.

**IWP** (4-byte float, array size: npixel1 x nscan1):
Ice water path.

**SWE** (4-byte float, array size: npixel1 x nscan1):
Snow water equivalent.

**SFR** (4-byte float, array size: npixel1 x nscan1):
Snow fall rate.

**C Structure Header file:**

```c
#ifndef _TK_1BASEMHS_H_
define _TK_1BASEMHS_H_

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
  short Year;
  signed char Month;
  signed char DayOfMonth;
  signed char Hour;
  signed char Minute;
  signed char Second;
  short MilliSecond;
  short DayOfYear;
```
double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L1BASEMHS_S1_
#define _L1BASEMHS_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[90];
    float Longitude[90];
    signed char isMissing;
    float sunGlintAngle[90];
    float solarBetaAngle;
    float timeSinceEclipseEntry;
    float tlePos[3];
    float tleVel[3];
    float sclat;
    float sclon;
    float scalt;
    double TimeTAI93;
    signed char sfcType[90];
    signed char orbitMode;
    float LZangle[90];
    float SZangle[90];
    float Chan1BT[90];
    float Chan2BT[90];
    float Chan3BT[90];
    float Chan4BT[90];
    float Chan5BT[90];
    float RR[90];
    float Snow[90];
    float IWP[90];
    float SWE[90];
    float SFR[90];
} L1BASEMHS_S1;

#endif

#ifndef _L1BASEMHS_BASEHEADER_
#define _L1BASEMHS_BASEHEADER_

5  STANDARD GPM PRODUCTS
typedef struct {
    int TLE_date[6];
    double TLE_time[1];
    double accf[270][5];
    double Bspace[5];
    float waveNumber[5];
} L1BASEMHS_BASEHEADER;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASEMHS_S1/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(90)
    REAL*4 Longitude(90)
    BYTE isMissing
    REAL*4 sunGlintAngle(90)
    REAL*4 solarBetaAngle
    REAL*4 timeSinceEclipseEntry
    REAL*4 tlePos(3)
    REAL*4 tleVel(3)
    REAL*4 sclat
    REAL*4 sclon
    REAL*4 scalt
    REAL*8 TimeTAI93
    BYTE sfcType(90)
    BYTE orbitMode
    REAL*4 LZangle(90)
REAL*4 SZangle(90)
REAL*4 Chan1BT(90)
REAL*4 Chan2BT(90)
REAL*4 Chan3BT(90)
REAL*4 Chan4BT(90)
REAL*4 Chan5BT(90)
REAL*4 RR(90)
REAL*4 Snow(90)
REAL*4 IWP(90)
REAL*4 SWE(90)
REAL*4 SFR(90)
END STRUCTURE

STRUCTURE /L1BASEMHS_BASEHEADER/>
  INTEGER*4 TLE_date(6)
  REAL*8 TLE_time(1)
  REAL*8 accf(5,270)
  REAL*8 Bspace(5)
  REAL*4 waveNumber(5)
END STRUCTURE

5.16 1BASESAPHIR - SAPHIR base

1BASESAPHIR contains brightness temperature from the SAPHIR passive microwave instrument flown on the Megha-Tropiques satellite. The channels are 183.1 +/- delta GHz, where delta = 0.2, 1.1, 2.8, 4.2, 6.8, 11.0.

Dimension definitions:
  nscan  var  Number of Swath 1 scans in the granule.
  nchannel  6  Number of channels.
  npixel  182  Number of pixels in one scan.
  three  3  Number of vectors.

Figure 262 through Figure 265 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the
Figure 262: Data Format Structure for 1BASESAPHIR, SAPHIR base
continued from last figure

Figure 263: Data Format Structure for 1BASESAPHIR, SAPHIR base

Figure 264: Data Format Structure for 1BASESAPHIR, baseHeader
same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**baseHeader** (Group)

**TLE_date** (4-byte integer, array size: 6):
TLE date time arrays.

**TLE_time** (8-byte float, array size: 1):
TLE time as in two line element.

**S1** (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

isMissing (1-byte integer, array size: nscan):
Missing scan flag.

timeTAI (8-byte float, array size: nscan):
Number of seconds since epoch time.

solarBetaAngle (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from
-59.0 to 59.0 degrees. Special values are defined as:
  -9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow.

**tlePos** (4-byte float, array size: three x nscan):
TLE satellite position. Values are in m. Special values are defined as:
  -9999.9 Missing value

**tleVel** (4-byte float, array size: three x nscan):
TLE satellite velocity. Values are in m/s. Special values are defined as:
  -9999.9 Missing value

**sclat** (4-byte float, array size: nscan):
Spacecraft latitude. Values range from -90 to 90.0 degree. Special values are defined as:
  -9999.9 Missing value

**sclon** (4-byte float, array size: nscan):
Spacecraft longitude. Values range from -180 to 180.0 degree. Special values are defined as:
  -9999.9 Missing value

**scalt** (4-byte float, array size: nscan):
Spacecraft altitude. Values range from 0 to 1000 km. Special values are defined as:
  -9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: npixel x nscan):
sunGlint angle.

**Latitude_Nadir** (4-byte float, array size: nscan):
Spacecraft latitude. Values range from -90 to 90.0 degree. Special values are defined as:
  -9999.9 Missing value

**Longitude_Nadir** (4-byte float, array size: nscan):
Spacecraft longitude. Values range from -180 to 180.0 degree. Special values are defined as:
  -9999.9 Missing value

**IncidenceAngle_Samples** (4-byte float, array size: npixel x nscan):
Incidence angle.

**Latitude** (4-byte float, array size: npixel x nscan):
Latitude.

**Longitude** (4-byte float, array size: npixel x nscan):
Longitude.

**Scan_Gain** (4-byte float, array size: nchannel x nscan):
Gain.

**Scan_Offset** (4-byte float, array size: nchannel x nscan):
Gain.
Scan_HotLoadTemperature (4-byte float, array size: nscan):
HotLoadTemperature

Scan_Number (2-byte integer, array size: nscan):
Scan number.

TB_Samples_S1 (4-byte float, array size: npixel x nscan):
TB_Samples_S1.

TB_Samples_S2 (4-byte float, array size: npixel x nscan):
TB_Samples_S2.

TB_Samples_S3 (4-byte float, array size: npixel x nscan):
TB_Samples_S3.

TB_Samples_S4 (4-byte float, array size: npixel x nscan):
TB_Samples_S4.

TB_Samples_S5 (4-byte float, array size: npixel x nscan):
TB_Samples_S5.

TB_Samples_S6 (4-byte float, array size: npixel x nscan):
TB_Samples_S6.

SAPHIR_QF_scan (2-byte unsigned integer, array size: nscan):

QF_Samples_S1 (2-byte unsigned integer, array size: npixel x nscan):
QF_Samples

QF_Samples_S2 (2-byte unsigned integer, array size: npixel x nscan):
QF_Samples

QF_Samples_S3 (2-byte unsigned integer, array size: npixel x nscan):
QF_Samples

QF_Samples_S4 (2-byte unsigned integer, array size: npixel x nscan):
QF_Samples

QF_Samples_S5 (2-byte unsigned integer, array size: npixel x nscan):
QF_Samples

QF_Samples_S6 (2-byte unsigned integer, array size: npixel x nscan):
QF_Samples

C Structure Header file:

```c
#ifndef _TK_1BASESAPHIR_H_
#define _TK_1BASESAPHIR_H_

#ifndef _SCANTIME_
#define _SCANTIME_
```

```c
```
typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L1BASESAPHIR_S1_
#define _L1BASESAPHIR_S1_

typedef struct {
    SCANTIME ScanTime;
    signed char isMissing;
    double timeTAI;
    float solarBetaAngle;
    float timeSinceEclipseEntry;
    float tlePos[3];
    float tleVel[3];
    float sclat;
    float sclon;
    float scalt;
    float sunGlintAngle_Samples[182];
    float Latitude_Nadir;
    float Longitude_Nadir;
    float IncidenceAngle_Samples[182];
    float Latitude[182];
    float Longitude[182];
    float Scan_Gain[6];
    float Scan_Offset[6];
    float Scan_HotLoadTemperature;
    short Scan_Number;
    float TB_Samples_S1[182];
    float TB_Samples_S2[182];
    float TB_Samples_S3[182];
    float TB_Samples_S4[182];
    float TB_Samples_S5[182];
} _L1BASESAPHIR_S1_;
float TB_Samples_S6[182];
unsigned short SAPHIR_QF_scan;
unsigned short QF_Samples_S1[182];
unsigned short QF_Samples_S2[182];
unsigned short QF_Samples_S3[182];
unsigned short QF_Samples_S4[182];
unsigned short QF_Samples_S5[182];
unsigned short QF_Samples_S6[182];
} L1BASESAPHIR_S1;
#endif
#endif
typedef struct {
    int TLE_date[6];
    double TLE_time[1];
} L1BASESAPHIR_BASEHEADER;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASESAPHIR_S1/
    RECORD /SCANTIME/ ScanTime
    BYTE isMissing
    REAL*8 timeTAI
    REAL*4 solarBetaAngle
5.17 1BASEATMS - ATMS base

1BASEATMS contains brightness temperature from the ATMS passive microwave instrument flown on the Suomi NPP satellite and JPSS satellites. ATMS is approximately AMSU-A plus MHS. Rotates 3 scans per 8 seconds. Input is SDR. 1BASEATMS = 1BATMS - PadBytes - BrightnessTemperatureFactors + isMissing + timeSinceEclipseEntry.
try + solarBetaAngle + sunGlintAngle + 6 times + tlePos + tleVel Data that occurs in 1BATMS 1, 2, 4 per ATMS granule appears per scan in 1BASEATMS, i.e., it is repeated. There is 1 swath with the following channels:

<table>
<thead>
<tr>
<th>Ch</th>
<th>GHz</th>
<th>Pol</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>23.8</td>
<td>QV</td>
</tr>
<tr>
<td>2</td>
<td>31.4</td>
<td>QV</td>
</tr>
<tr>
<td>3</td>
<td>50.3</td>
<td>QH</td>
</tr>
<tr>
<td>4</td>
<td>51.76</td>
<td>QH</td>
</tr>
<tr>
<td>5</td>
<td>52.8</td>
<td>QH</td>
</tr>
<tr>
<td>6</td>
<td>53.596±0.115</td>
<td>QH</td>
</tr>
<tr>
<td>7</td>
<td>54.4</td>
<td>QH</td>
</tr>
<tr>
<td>8</td>
<td>54.94</td>
<td>QH</td>
</tr>
<tr>
<td>9</td>
<td>55.5</td>
<td>QH</td>
</tr>
<tr>
<td>10</td>
<td>fo = 57.29</td>
<td>QH</td>
</tr>
<tr>
<td>11</td>
<td>fo±0.3222±0.217</td>
<td>QH</td>
</tr>
<tr>
<td>12</td>
<td>fo±0.3222±0.048</td>
<td>QH</td>
</tr>
<tr>
<td>13</td>
<td>fo±0.3222±0.022</td>
<td>QH</td>
</tr>
<tr>
<td>14</td>
<td>fo±0.3222±0.010</td>
<td>QH</td>
</tr>
<tr>
<td>15</td>
<td>fo±0.3222±0.0045</td>
<td>QH</td>
</tr>
<tr>
<td>16</td>
<td>88.2</td>
<td>QV</td>
</tr>
<tr>
<td>17</td>
<td>165.5</td>
<td>QH</td>
</tr>
<tr>
<td>18</td>
<td>183.31+/-7</td>
<td>QH</td>
</tr>
<tr>
<td>19</td>
<td>183.31+/-4.5</td>
<td>QH</td>
</tr>
<tr>
<td>20</td>
<td>183.31+/-3</td>
<td>QH</td>
</tr>
<tr>
<td>21</td>
<td>183.31+/-1.8</td>
<td>QH</td>
</tr>
<tr>
<td>22</td>
<td>183.31+/-1</td>
<td>QH</td>
</tr>
</tbody>
</table>

Note on geolocation and 1C swaths:
The BeamLatitude and BeamLongitude in 1BASEATMS have a band dimension of 5. Lat and lon is for channels 1,2,3,16,17. Each 1C swath will contain one band:

<table>
<thead>
<tr>
<th>1C swath</th>
<th>Band</th>
<th>IEEE GHz</th>
<th>Ch geo</th>
<th>Chs in band</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>K</td>
<td>18-26.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>A(Ka)</td>
<td>26.5-40</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>V</td>
<td>50-75</td>
<td>3</td>
<td>3-15</td>
</tr>
<tr>
<td>4</td>
<td>W</td>
<td>75-110</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>G</td>
<td>110-300</td>
<td>17</td>
<td>17-22</td>
</tr>
</tbody>
</table>

More detailed information on some variables may be found in the document JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats.

Dimension definitions:
nscan  var Number of scans in the granule.
nchannel  22 Number of channels.
nbeam  96 Number of beams in one scan.
nband  5 Number of bands (K,A,V,W,G).
vecsize  3 Vector size.
three  3 Number of vectors.
seven  7 Number of dimensions in time array.

Figure 266 through Figure 269 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**S1** (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value
Figure 266: Data Format Structure for 1BASEATMS, ATMS base
Figure 267: Data Format Structure for 1BASEATMS, ATMS base
continued from last figure

Figure 268: Data Format Structure for 1BASEATMS, ATMS base

Figure 269: Data Format Structure for 1BASEATMS, ScanTime
Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
   -99  Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
   -99  Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
   -99  Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
   -99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
   -99  Missing value

MillisSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
   -9999  Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
   -9999  Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
   -9999.9  Missing value

isMissing (1-byte integer, array size: nscan):
Missing scan flag.

sclat (4-byte float, array size: nscan):
The geodedic latitude of the spacecraft at the scan mid-time. Values range from -90 to
90.0 degrees. Special values are defined as:
   -9999.9  Missing value

sclon (4-byte float, array size: nscan):
The geodedic longitude of the spacecraft at the scan mid-time. Values range from -180
to 180.0 degrees. Special values are defined as:
   -9999.9  Missing value

scalt (4-byte float, array size: nscan):
The altitude of the spacecraft at the scan mid-time. Values range from 0 to 1000 km.
Special values are defined as:
   -9999.9  Missing value
solarBetaAngle (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (4-byte float, array size: nbeam x nscan):
sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection.

timeSinceEclipseEntry (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow.

BeamTime (8-byte float, array size: nbeam x nscan):
The time in IET (seconds since 1958-01-01 00:00:00) of the end of the view period for this observation.

BrightnessTemperature (4-byte float, array size: nchannel x nbeam x nscan):
Calibrated scene brightness temperature.

GainCalibration (4-byte float, array size: nchannel x nscan):
Gain factor used in calibrating earth scene brightness temperature.

InstrumentMode (2-byte unsigned integer, array size: nscan):
Instrument mode.

NEdTCold (4-byte float, array size: nchannel x nscan):
Noise-equivalent delta temperature while viewing cold space.

NEdTWarm (4-byte float, array size: nchannel x nscan):
Noise-equivalent delta temperature while viewing warm target.

QF1_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF2_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF3_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF4_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF5_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.
QF6_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data
Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF7_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data
Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF8_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data
Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF9_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data
Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF10_GRAN_HEALTHSTATUS (1-byte char, array size: nscan):
Out of range quality flag for 8 second health and status packet. See JPSS Common Data
Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF11_GRAN_QUADRATICCORRECTION (1-byte char, array size: nscan):
Quadratic correction applied to the radiometric transfer function for non-linearity cor-
rection. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF12_SCAN_KAVPR_TCONVERR (1-byte char, array size: nscan):
If a divide-by-zero condition exists, or if computation loop fails to converge in the temper-
ature computations for the 8 KAV PRTs, the condition is flagged by the corresponding
bit in the flag to indicate which PRT has failed. See JPSS Common Data Format Control
Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF13_SCAN_WGPRTCONVERR (1-byte char, array size: nscan):
If a divide-by-zero condition exists, or if computation loop fails to converge in the tem-
perature computations for the 7 WG PRTs, the condition is flagged by the corresponding
bit in the flag to indicate which PRT has failed. See JPSS Common Data Format Control
Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF14_SCAN_SHELFPR_TCONVERR (1-byte char, array size: nscan):
If a divide-by-zero condition exists, or if computation loop fails to converge in the tem-
perature computations for the 4 receiver shelf (KKa, V, W and G) PRTs, the condition is
flagged by the corresponding bit in the flag to indicate which PRT has failed. See JPSS
Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF15_SCAN_KAVPRTTEMPLIMIT (1-byte char, array size: nscan):
Each of the 8 KAV PRT temperatures is checked against a lower limit and an upper limit.
Out of range conditions are flagged by the corresponding bit in the flag to indicate which
PRT has failed the test. See JPSS Common Data Format Control Book - Vol III Sensor
Data Record (SDR)/TDR Formats for details.
QF16_SCAN_WGPR_TEMPLIMIT (1-byte char, array size: nscan):
Each of the 7 WG PRT temperatures is checked against a lower limit and an upper limit.
Out of range conditions are flagged by the corresponding bit in the flag to indicate which
PRT has failed the test. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF17_SCAN_KAVPR_T_TEMPCONSISTENCY (1-byte char, array size: nscan):
The 8 KAV PRT temperatures are checked against each other for consistency. The check
failure shall be flagged by the corresponding bit in the flag to indicate which PRT has
failed the test. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF18_SCAN_WGPR_T_TEMPCONSISTENCY (1-byte char, array size: nscan):
The 7 WG PRT temperatures are checked against each other for consistency. The check
failure shall be flagged by the corresponding bit in the flag to indicate which PRT has
failed the test. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF19_SCAN_ATMSSDR (1-byte char, array size: nscan):
Scan level quality flag. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF20_ATMSSDR (1-byte char, array size: nchannel x nscan):
Scan level quality flag per channel. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF21_ATMSSDR (1-byte char, array size: nchannel x nscan):
Out of range space and blackbody view quality flag. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

QF22_ATMSSDR (1-byte char, array size: nchannel x nscan):
Space and blackbody view quality flag. See JPSS Common Data Format Control Book - Vol III Sensor Data Record (SDR)/TDR Formats for details.

BeamLatitude (4-byte float, array size: nband x nbeam x nscan):
Latitude of individual beam position centers (channels 1, 2, 3, 16, 17).

BeamLongitude (4-byte float, array size: nband x nbeam x nscan):
Longitude of individual beam position centers (channels 1, 2, 3, 16, 17).

Height (4-byte float, array size: nbeam x nscan):
Ellipsoid-Geoid separation.

Latitude (4-byte float, array size: nbeam x nscan):
Latitude of channel 17 beam position center

Longitude (4-byte float, array size: nbeam x nscan):
Longitude of channel 17 beam position center

MidTime (8-byte float, array size: nscan):
Mid time of scan in IET (seconds since 1958-01-01 00:00:00)
QF1_ATMSSDRGEO (1-byte char, array size: nscan):
Attitude and Ephemeris availability status

<table>
<thead>
<tr>
<th>value</th>
<th>meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal</td>
</tr>
<tr>
<td>1</td>
<td>missing data less than or equal to small gap</td>
</tr>
<tr>
<td>2</td>
<td>missing data larger than small gap, but less than</td>
</tr>
<tr>
<td>3</td>
<td>missing data larger than or equal to granule boundary</td>
</tr>
</tbody>
</table>

SCAttitude (4-byte float, array size: vecsize x nscan):
Spacecraft attitude with respect to Geodetic Reference Frame Coordinates (roll, pitch, yaw) at the mid-time of scan.

SCPosition (4-byte float, array size: vecsize x nscan):
Spacecraft position in Earth Centered Rotating (ECR) Coordinates (X, Y, Z) at the mid-time of scan.

SCVelocity (4-byte float, array size: vecsize x nscan):
Spacecraft velocity in Earth Centered Rotating (ECR) Coordinates (dx/dt, dy/dt, dz/dt) at the mid-time of scan.

SatelliteAzimuthAngle (4-byte float, array size: nbeam x nscan):
Azimuth angle (measured clockwise positive from North) to satellite at the geolocated beam position center.

SatelliteRange (4-byte float, array size: nbeam x nscan):
Line of sight distance from the ellipsoid intersection to the satellite.

incidenceAngle_23 (4-byte float, array size: nbeam x nscan):
Earth incidence angle 23 GHz.

incidenceAngle_31 (4-byte float, array size: nbeam x nscan):
Earth incidence angle 31 GHz.

incidenceAngle_50 (4-byte float, array size: nbeam x nscan):
Earth incidence angle 50 GHz.

incidenceAngle_88 (4-byte float, array size: nbeam x nscan):
Earth incidence angle 88 GHz.

incidenceAngle_165 (4-byte float, array size: nbeam x nscan):
Earth incidence angle 165 GHz.

SolarAzimuthAngle (4-byte float, array size: nbeam x nscan):
Azimuth angle (measured clockwise positive from North) of sun at the geolocated beam position center.

SolarZenithAngle (4-byte float, array size: nbeam x nscan):
Zenith angle to sun at the geolocated beam position center.
**StartTime** (8-byte float, array size: nscan):
Starting time of scan in IET (seconds since 1958-01-01 00:00:00).

### C Structure Header file:

```c
#ifndef _TK_1BASEATMS_H_
define _TK_1BASEATMS_H_

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
  short Year;
  signed char Month;
  signed char DayOfMonth;
  signed char Hour;
  signed char Minute;
  signed char Second;
  short MilliSecond;
  short DayOfYear;
  double SecondOfDay;
} SCANTIME;
#endif
#endif

#ifndef _L1BASEATMS_S1_
define _L1BASEATMS_S1_

typedef struct {
  SCANTIME ScanTime;
  signed char isMissing;
  float sclat;
  float sclon;
  float scalt;
  float solarBetaAngle;
  float sunGlintAngle[96];
  float timeSinceEclipseEntry;
  double BeamTime[96];
  float BrightnessTemperature[96][22];
  float GainCalibration[22];
  unsigned short InstrumentMode;
  float NEdTCold[22];
  float NEdTTemp[22];
  unsigned char QF1_GRAN_HEALTHSTATUS;
```
unsigned char QF2_GRAN_HEALTHSTATUS;
unsigned char QF3_GRAN_HEALTHSTATUS;
unsigned char QF4_GRAN_HEALTHSTATUS;
unsigned char QF5_GRAN_HEALTHSTATUS;
unsigned char QF6_GRAN_HEALTHSTATUS;
unsigned char QF7_GRAN_HEALTHSTATUS;
unsigned char QF8_GRAN_HEALTHSTATUS;
unsigned char QF9_GRAN_HEALTHSTATUS;
unsigned char QF10_GRAN_HEALTHSTATUS;
unsigned char QF11_GRAN_QUADRATICCORRECTION;
unsigned char QF12_SCAN_KAVPRTCONVERR;
unsigned char QF13_SCAN_WGPRTCONVERR;
unsigned char QF14_SCAN_SHELFPRTCONVERR;
unsigned char QF15_SCAN_KAVPRTTEMPLIMIT;
unsigned char QF16_SCAN_WGPRTTEMPLIMIT;
unsigned char QF17_SCAN_KAVPRTTEMPCONSISTENCY;
unsigned char QF18_SCAN_WGPRTTEMPCONSISTENCY;
unsigned char QF19_SCAN_ATMSSDR;
unsigned char QF20_ATMSSDR[22];
unsigned char QF21_ATMSSDR[22];
unsigned char QF22_ATMSSDR[22];
float BeamLatitude[96][5];
float BeamLongitude[96][5];
float Height[96];
float Latitude[96];
float Longitude[96];
double MidTime;
unsigned char QF1_ATMSSDRGEO;
float SCAttitude[3];
float SCPPosition[3];
float SCVelocity[3];
float SatelliteAzimuthAngle[96];
float SatelliteRange[96];
float incidenceAngle_23[96];
float incidenceAngle_31[96];
float incidenceAngle_50[96];
float incidenceAngle_88[96];
float incidenceAngle_165[96];
float SolarAzimuthAngle[96];
float SolarZenithAngle[96];
double StartTime;
} L1BASEATMS_S1;
Fortran Structure Header file:

STRUCTURE /SCANTIME/
   INTEGER*2 Year
   BYTE Month
   BYTE DayOfMonth
   BYTE Hour
   BYTE Minute
   BYTE Second
   INTEGER*2 MilliSecond
   INTEGER*2 DayOfYear
   REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BASEATMS_S1/
   RECORD /SCANTIME/ ScanTime
   BYTE isMissing
   REAL*4 sclat
   REAL*4 sclon
   REAL*4 scalt
   REAL*4 solarBetaAngle
   REAL*4 sunGlintAngle(96)
   REAL*4 timeSinceEclipseEntry
   REAL*8 BeamTime(96)
   REAL*4 BrightnessTemperature(22,96)
   REAL*4 GainCalibration(22)
   INTEGER*2 InstrumentMode
   REAL*4 NEdTCold(22)
   REAL*4 NEdTWarm(22)
   CHARACTER QF1_GRAN_HEALTHSTATUS
   CHARACTER QF2_GRAN_HEALTHSTATUS
   CHARACTER QF3_GRAN_HEALTHSTATUS
   CHARACTER QF4_GRAN_HEALTHSTATUS
   CHARACTER QF5_GRAN_HEALTHSTATUS
   CHARACTER QF6_GRAN_HEALTHSTATUS
   CHARACTER QF7_GRAN_HEALTHSTATUS
   CHARACTER QF8_GRAN_HEALTHSTATUS
   CHARACTER QF9_GRAN_HEALTHSTATUS
   CHARACTER QF10_GRAN_HEALTHSTATUS
The Level-1B GMI Product, 1BGMI, “GMI Brightness Temperatures,” is written as a multi-Swath Structure. Swath S1 has channels 1-9: 10V 10H 19V 19H 23V 37V 37H 89V 89H. Swath S2 has channels 10-13: 166V 166H 183+/−3V 183+/−8V. The following sections describe the structure and contents of the format.

Dimension definitions:
Figure 270: Data Format Structure for IBGMI, GMI Brightness Temperatures

nscan  var  Number of scans in the granule.
nchan1  9  Number of channels in Swath 1.
nchan2  4  Number of channels in Swath 2.
nfreq1  5  Number of frequencies in Swath 1.
nfreq2  2  Number of frequencies in Swath 2.
npix1  221  Number of pixels in Swath 1.
npix2  221  Number of pixels in Swath 2.
ncolds1  85  Maximum number of cold samples in Swath 1.
ncolds2  85  Maximum number of cold samples in Swath 2.
nhots1  65  Maximum number of hot samples in Swath 1.
nhots2  65  Maximum number of hot samples in Swath 2.
ntherm  11  Number of hot load thermisters.
LNL  2  Linear and non-linear.
nsamt  4  Number of sample types. The types are: total science GSDR, earth-view, hot load, cold sky.
ntach  32  Number of tachometer readings.
GMIxyz  3  x, y, z components in GMI instrument coordinate system.

Figure 270 through Figure 286 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 271: Data Format Structure for 1BGMI, S1
Figure 272: Data Format Structure for 1BGM1, S2

Figure 273: Data Format Structure for 1BGM1, S1, ScanTime
Figure 274: Data Format Structure for 1BGMI, S1, scanStatus

Figure 275: Data Format Structure for 1BGMI, S1, sampleHeader
Figure 276: Data Format Structure for 1BGMI, S1, navigation
5.18 1BGMI - GMI Brightness Temperatures

Figure 277: Data Format Structure for 1BGMI, S1, calibration

Figure 278: Data Format Structure for 1BGMI, S1, calCounts
Figure 279: Data Format Structure for 1BGMI, S1, sunData

Figure 280: Data Format Structure for 1BGMI, S2, ScanTime
Figure 281: Data Format Structure for 1BGMI, S2, scanStatus
**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

S1 (Swath)

**S1_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S1)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- 9999    Missing value
5.18 1BGMI - GMI Brightness Temperatures

Figure 283: Data Format Structure for 1BGMI, S2, navigation
5 STANDARD GPM PRODUCTS

- **calibration**
  - hotLoadTemp 4 bytes Array: nchan2 x nscan
  - coldSkyTemp 4 bytes Array: nchan2 x nscan
  - onOrbitNonLinearity 4 bytes Array: nchan2 x nscan
  - derivedNonLinearity 4 bytes Array: nchan2 x nscan
  - meanHotLoadCount 2 bytes Array: nchan2 x nscan
  - meanHotLoadCntnDiode 2 bytes Array: nchan2 x nscan
  - meanColdSkyCount 2 bytes Array: nchan2 x nscan
  - meanColdSkyCntnDiode 2 bytes Array: nchan2 x nscan
  - diodeCoupledTemp 4 bytes Array: nchan2 x nscan
  - gain 4 bytes Array: LNL x nchan2 x nscan
  - offset 4 bytes Array: LNL x nchan2 x nscan
  - nonLinearGain 4 bytes Array: nchan2 x nscan
  - calibrationQCflag 2 bytes Array: nscan
  - diodeFlag 2 bytes Array: nscan
  - receiverTemp 4 bytes Array: nchan2 x nscan
  - receiverGain 4 bytes Array: nchan2 x nscan

Figure 284: Data Format Structure for 1BGMI, S2, calibration

- **calCounts**
  - hotLoadThermisterTemp 4 bytes Array: ntherm x nchan2 x nscan
  - hotLoadReading 2 bytes Array: nhots2 x nchan2 x nscan
  - coldLoadReading 2 bytes Array: ncolds2 x nchan2 x nscan
  - hotLoadnDiodeReading 2 bytes Array: nhots2 x nchan2 x nscan
  - coldLoadnDiodeReading 2 bytes Array: ncolds2 x nchan2 x nscan

Figure 285: Data Format Structure for 1BGMI, S2, calCounts
### Sun Data

- **solarBetaAngle** (4 bytes, array size: nscan)
- **phaseFromOrbitMidnight** (4 bytes, array size: nscan)
- **sunEarthSeparation** (4 bytes, array size: nscan)
- **earthAngularRadius** (4 bytes, array size: nscan)
- **phaseOfEclipseExit** (4 bytes, array size: nscan)
- **orbitRate** (4 bytes, array size: nscan)
- **timeSinceEclipseEntry** (4 bytes, array size: nscan)
- **sunVectorInBodyFrame** (4 bytes, array size: 3 x nscan)

![Figure 286: Data Format Structure for 1BGMI, S2, sunData](image)

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
- 99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
- 99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
- 99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
- 99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
- 99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
- 9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
- 9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
- 9999.9 Missing value

**Latitude** (4-byte float, array size: npix1 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**Longitude** (4-byte float, array size: npix1 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scanStatus** (Group in S1)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)). The non-routine situations follow:
### 5.18 1BGMI - GMI Brightness Temperatures

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$ the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit $7 = 1$ and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit $7 = 0$ and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be needed.
useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector ($v$) from the satellite forward direction of motion, measured clockwise facing down. We define $v$ in the same direction as the spacecraft axis $+X$, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>-9999.9</td>
<td>Missing value</td>
</tr>
</tbody>
</table>
sampleHeader (Group in S1)

blanking (1-byte integer, array size: nscan):

Value of 0 = Table 0 used for hot and cold samples, 
   No blanking
Value of 1 = Table 1 used for hot and cold samples, 
   Blanking on both sides
Value of 2 = Table 2 used for hot and cold samples, 
   Blanking on begin side
Value of 3 = Table 3 used for hot and cold samples, 
   Blanking on end side

earthViewFirstSample (2-byte integer, array size: nscan):
Sample number of the first earth view. Values range from 0 to 512. Special values are 
defined as:
   -9999 Missing value

sampleNumber (2-byte integer, array size: nsamt x nchan1 x nscan):
Number of valid samples in scan. Values range from 0 to 512. Special values are defined 
as:
   -9999 Missing value

tachSeconds (4-byte unsigned integer, array size: ntach x nscan):
Tachometer seconds. Special values are defined as:
   0 Missing value

tachMicroSeconds (2-byte unsigned integer, array size: ntach x nscan):
Tachometer microseconds. Special values are defined as:
   0 Missing value

navigation (Group in S1)

scPos (4-byte float, array size: XYZ x nscan):
The position vector(m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coor-
dinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan 
period). Values range from -10000000 to 10000000 m. Special values are defined as:
   -9999.9 Missing value

cvVel (4-byte float, array size: XYZ x nscan):
The velocity vector (m/s\(^{-1}\)) of the spacecraft in ECEF Coordinates at the Scan mid-Time. 
Values range from -10000000 to 10000000 m/s. Special values are defined as:
   -9999.9 Missing value
scLat (4-byte float, array size: nscan):
The geodetic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geodetic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value
**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**calibration** (Group in S1)

**hotLoadTemp** (4-byte float, array size: nchan1 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value
coldSkyTemp (4-byte float, array size: nchan1 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

onOrbitNonLinearity (4-byte float, array size: nchan1 x nscan):
The on Orbit Non-Linearity. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

derivedNonLinearity (4-byte float, array size: nchan1 x nscan):
The derived Non-Linearity. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

meanHotLoadCount (2-byte unsigned integer, array size: nchan1 x nscan):
The mean Hot Load Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value

meanHotLoadCntnDiode (2-byte unsigned integer, array size: nchan1 x nscan):
The mean Hot Load Count Plus Noise Diode. Values range from 0 to 15. Special values are defined as:
65535 Missing value

meanColdSkyCount (2-byte unsigned integer, array size: nchan1 x nscan):
The mean Cold Sky Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value

meanColdSkyCntnDiode (2-byte unsigned integer, array size: nchan1 x nscan):
The mean Cold Sky Count Plus Noise Diode. Values range from 0 to 15. Special values are defined as:
65535 Missing value

diodeCoupledTemp (4-byte float, array size: nchan1 x nscan):
The diode Coupled Temp. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

gain (4-byte float, array size: LNL x nchan1 x nscan):
Automatic gain control. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

offset (4-byte float, array size: LNL x nchan1 x nscan):
Offset. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

nonLinearGain (4-byte float, array size: nchan1 x nscan):
The nonlinear gain. Special values are defined as:
-9999.9 Missing value

calibrationQCflag (2-byte integer, array size: nscan):
calibrationQCflag. Values range from 0 to 15. Special values are defined as:
-9999 Missing value
**diodeFlag** (2-byte integer, array size: nscan):
Diode flag. If diodeFlag = 1, use LoadPlusDiodeReading If diodeFlag = 0, use LoadReading Values range from 0 to 1 counts. Special values are defined as:
-9999 Missing value

**receiverTemp** (4-byte float, array size: nchan1 x nscan):
The receiver temperature. Special values are defined as:
-9999.9 Missing value

**receiverGain** (4-byte float, array size: nchan1 x nscan):
The receiver gain. Special values are defined as:
-9999.9 Missing value

**moonVectorInstFrame** (4-byte float, array size: GMIxyz x nscan):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
-9999.9 Missing value

**calCounts** (Group in S1)

**hotLoadThermisterTemp** (4-byte float, array size: ntherm x nchan1 x nscan):
Hot Load Thermister Temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**hotLoadReading** (2-byte unsigned integer, array size: nhots1 x nchan1 x nscan):
Hot Load Reading. Values range from 0 to 15 counts. Special values are defined as:
0 Missing value

**coldLoadReading** (2-byte unsigned integer, array size: ncolds1 x nchan1 x nscan):
Cold Load Reading. Values range from 0 to 15 counts. Special values are defined as:
0 Missing value

**hotLoadnDiodeReading** (2-byte unsigned integer, array size: nhots1 x nchan1 x nscan):
Hot Load Plus Diode Reading. Values range from 0 to 15 counts. Special values are defined as:
0 Missing value

**coldLoadnDiodeReading** (2-byte unsigned integer, array size: ncolds1 x nchan1 x nscan):
Cold Load Plus Diode Reading. Values range from 0 to 15 counts. Special values are defined as:
0 Missing value
sunData (Group in S1)

solarBetaAngle (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nscan):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special
values are defined as:
-9999.9  Missing value

**incidenceAngle** (4-byte float, array size: npix1 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**satAzimuthAngle** (4-byte float, array size: npix1 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**solarZenAngle** (4-byte float, array size: npix1 x nscan):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**solarAzimuthAngle** (4-byte float, array size: npix1 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**sunGlintAngle** (4-byte float, array size: npix1 x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**Tb** (4-byte float, array size: nchan1 x npix1 x nscan):
Earth view brightness temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**RFIFlag** (2-byte integer, array size: nfreq1 x npix1 x nscan):
Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:
0:  Not affected by RFI.
1:  Affected by RFI with X-cal filter.
2:  Affected by RFI with RSS filter.
3-7: Spare
-9999: Missing

S2 (Swath)

S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value
SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npix2 x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npix2 x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in S2)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit =
1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other
bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

g eoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in
dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be
useful. A zero integer value indicates usual geolocation. A non-zero value broken down
into the following bit flags indicates the following, where bit 0 is the least significant bit
(i.e., if bit \( i = 1 \) and other bits = 0 the unsigned integer value is \( 2^{**i} \)):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector \( (v) \) from the satellite forward direction of mo-
tion, measured clockwise facing down. We define \( v \) in the same direction as the spacecraft
axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit
is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is
good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit
in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**acsModeMidScan** (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):
Status of the GMI instrument.

<table>
<thead>
<tr>
<th>Bit Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>1 Spinup Status (0=ON, 1=OFF)</td>
</tr>
</tbody>
</table>
FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of 
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the 
spacecraft is halfway through granule 10 and starting the descending half of the granule. 
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

sampleHeader (Group in S2)

blanking (1-byte integer, array size: nscan):

Value of 0 = Table 0 used for hot and cold samples, 
No blanking
Value of 1 = Table 1 used for hot and cold samples, 
Blanking on both sides
Value of 2 = Table 2 used for hot and cold samples, 
Blanking on begin side
Value of 3 = Table 3 used for hot and cold samples, 
Blanking on end side

earthViewFirstSample (2-byte integer, array size: nscan):
Sample number of the first earth view. Values range from 0 to 512. Special values are 
defined as:
-9999 Missing value

sampleNumber (2-byte integer, array size: nsamt x nchan2 x nscan):
Number of valid samples in scan. Values range from 0 to 512. Special values are defined 
as:
-9999 Missing value

tachSeconds (4-byte unsigned integer, array size: ntach x nscan):
Tachometer seconds. Special values are defined as:
0 Missing value

tachMicroSeconds (2-byte unsigned integer, array size: ntach x nscan):
Tachometer microseconds. Special values are defined as:
0 Missing value

navigation (Group in S2)
**scPos** (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector \((m s^{-1})\) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value
**calibration** (Group in S2)

**hotLoadTemp** (4-byte float, array size: nchan2 x nscan):
The mean physical temperature for the temperature sensors attached to the hot load. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**coldSkyTemp** (4-byte float, array size: nchan2 x nscan):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**onOrbitNonLinearity** (4-byte float, array size: nchan2 x nscan):
The on Orbit Non-Linearity. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**derivedNonLinearity** (4-byte float, array size: nchan2 x nscan):
The derived Non-Linearity. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**meanHotLoadCount** (2-byte unsigned integer, array size: nchan2 x nscan):
The mean Hot Load Count. Values range from 0 to 15. Special values are defined as:
65535  Missing value

**meanHotLoadCntnDiode** (2-byte unsigned integer, array size: nchan2 x nscan):
The mean Hot Load Count Plus Noise Diode. Values range from 0 to 15. Special values are defined as:
65535  Missing value

**meanColdSkyCount** (2-byte unsigned integer, array size: nchan2 x nscan):
The mean Cold Sky Count. Values range from 0 to 15. Special values are defined as:
65535  Missing value

**meanColdSkyCntnDiode** (2-byte unsigned integer, array size: nchan2 x nscan):
The mean Cold Sky Count Plus Noise Diode. Values range from 0 to 15. Special values are defined as:
65535  Missing value

**diodeCoupledTemp** (4-byte float, array size: nchan2 x nscan):
The diode Coupled Temp. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

**gain** (4-byte float, array size: LNL x nchan2 x nscan):
Automatic gain control. Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value
offset (4-byte float, array size: LNL x nchan2 x nscan):
Offset. Values range from 0 to 400 K. Special values are defined as:
   -9999.9 Missing value

nonLinearGain (4-byte float, array size: nchan2 x nscan):
The nonlinear gain. Special values are defined as:
   -9999.9 Missing value

calibrationQCflag (2-byte integer, array size: nscan):
calibrationQCflag. Values range from 0 to 15. Special values are defined as:
   -9999 Missing value

diodeFlag (2-byte integer, array size: nscan):
Diode flag. If diodeFlag = 1, use LoadPlusDiodeReading If diodeFlag = 0, use Load-
Reading. Values range from 0 to 1 counts. Special values are defined as:
   -9999 Missing value

receiverTemp (4-byte float, array size: nchan2 x nscan):
The receiver temperature. Special values are defined as:
   -9999.9 Missing value

receiverGain (4-byte float, array size: nchan2 x nscan):
The receiver gain. Special values are defined as:
   -9999.9 Missing value

moonVectorInstFrame (4-byte float, array size: GMIxyz x nscan):
The x, y, z components of the moon vector in the GMI instrument coordinate system.
Values are in counts. Special values are defined as:
   -9999.9 Missing value

calCounts (Group in S2)

hotLoadThermisterTemp (4-byte float, array size: ntherm x nchan2 x nscan):
Hot Load Thermister Temperature. Values range from 0 to 400 K. Special values are
defined as:
   -9999.9 Missing value

hotLoadReading (2-byte unsigned integer, array size: nhots2 x nchan2 x nscan):
Hot Load Reading. Values range from 0 to 15 counts. Special values are defined as:
   0 Missing value

coldLoadReading (2-byte unsigned integer, array size: ncolds2 x nchan2 x nscan):
Cold Load Reading. Values range from 0 to 15 counts. Special values are defined as:
   0 Missing value

hotLoadnDiodeReading (2-byte unsigned integer, array size: nhots2 x nchan2 x nscan):
Hot Load Plus Diode Reading. Values range from 0 to 15 counts. Special values are
defined as:
  0   Missing value

**coldLoadnDiodeReading** (2-byte unsigned integer, array size: ncolds2 x nchan2 x nsca)
Cold Load Plus Diode Reading. Values range from 0 to 15 counts. Special values are defined as:
  0   Missing value

**sunData** (Group in S2)

**solarBetaAngle** (4-byte float, array size: nscan):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
  -9999.9  Missing value

**phaseFromOrbitMidnight** (4-byte float, array size: nscan):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
  -9999.9  Missing value

**sunEarthSeparation** (4-byte float, array size: nscan):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
  -9999.9  Missing value

**earthAngularRadius** (4-byte float, array size: nscan):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
  -9999.9  Missing value

**phaseOfEclipseExit** (4-byte float, array size: nscan):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
  -9999.9  Missing value

**orbitRate** (4-byte float, array size: nscan):
The instantaneous angular rate of the spacecraft around the orbit. Values range from
0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value

incidenceAngle (4-byte float, array size: npix2 x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

satAzimuthAngle (4-byte float, array size: npix2 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarZenAngle (4-byte float, array size: npix2 x nscan):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarAzimuthAngle (4-byte float, array size: npix2 x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (4-byte float, array size: npix2 x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsiod-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

Tb (4-byte float, array size: nchan2 x npix2 x nscan):
Earth view brightness temperature. Values range from 0 to 400 K. Special values are
defined as:
-9999.9 Missing value

**RFIFlag** (2-byte integer, array size: nfreq2 x npix2 x nscan):

Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: Not affected by RFI.
1: Affected by RFI with X-cal filter.
2: Affected by RFI with RSS filter.
3-7: Spare
-9999: Missing

C Structure Header file:

```c
#ifndef _TK_1BGMI_H_
#define _TK_1BGMI_H_

#ifndef _L1BGMI_S2_SUNDATA_
#define _L1BGMI_S2_SUNDATA_

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
    float sunVectorInBodyFrame[3];
} L1BGMI_S2_SUNDATA;
#endif

#ifndef _L1BGMI_S2_CALCOUNTS_
#define _L1BGMI_S2_CALCOUNTS_

typedef struct {
    float hotLoadThermisterTemp[4][11];
    unsigned short hotLoadReading[4][65];
    unsigned short coldLoadReading[4][85];
```
unsigned short hotLoadnDiodeReading[4][65];
unsigned short coldLoadnDiodeReading[4][85];
} L1BGMI_S2_CALCOUNTS;

#endif

#ifndef _L1BGMI_S2_CALIBRATION_
#define _L1BGMI_S2_CALIBRATION_

typedef struct {
   float hotLoadTemp[4];
   float coldSkyTemp[4];
   float onOrbitNonLinearity[4];
   float derivedNonLinearity[4];
   unsigned short meanHotLoadCount[4];
   unsigned short meanHotLoadCntnDiode[4];
   unsigned short meanColdSkyCount[4];
   unsigned short meanColdSkyCntnDiode[4];
   float diodeCoupledTemp[4];
   float gain[4][2];
   float offset[4][2];
   float nonLinearGain[4];
   short calibrationQCflag;
   short diodeFlag;
   float receiverTemp[4];
   float receiverGain[4];
} L1BGMI_S2_CALIBRATION;

#endif

#ifndef _L1BGMI_S2_SAMPLEHEADER_
#define _L1BGMI_S2_SAMPLEHEADER_

typedef struct {
   signed char blanking;
   short earthViewFirstSample;
   short sampleNumber[4][4];
   unsigned int tachSeconds[32];
   unsigned short tachMicroSeconds[32];
} L1BGMI_S2_SAMPLEHEADER;

#endif
#ifndef _L1BGMI_S2_SCANSTATUS_
define _L1BGMI_S2_SCANSTATUS_

typedef struct {
signed char dataQuality;
signed char missing;
signed char modeStatus;
short geoError;
short geoWarning;
short SCorientation;
short pointingStatus;
signed char acsModeMidScan;
signed char targetSelectionMidScan;
signed char operationalMode;
double FractionalGranuleNumber;
} L1BGMI_S2_SCANSTATUS;
#endif

#endif

#ifndef _L1BGMI_S2_
define _L1BGMI_S2_

typedef struct {
SCANTIME ScanTime;
float Latitude[221];
float Longitude[221];
L1BGMI_S2_SCANSTATUS scanStatus;
L1BGMI_S2_SAMPLEHEADER sampleHeader;
NAVIGATION navigation;
L1BGMI_S2_CALIBRATION calibration;
float moonVectorInstFrame[3];
L1BGMI_S2_CALCOUNTS calCounts;
L1BGMI_S2_SUNDATA sunData;
float incidenceAngle[221];
float satAzimuthAngle[221];
float solarZenAngle[221];
float solarAzimuthAngle[221];
float sunGlintAngle[221];
float Tb[221][4];
short RFIFlag[221][2];
} L1BGMI_S2;
#endif
#ifndef _L1BGMI_S1_SUNDATA_
define _L1BGMI_S1_SUNDATA_

typedef struct {
  float solarBetaAngle;
  float phaseFromOrbitMidnight;
  float sunEarthSeparation;
  float earthAngularRadius;
  float phaseOfEclipseExit;
  float orbitRate;
  float timeSinceEclipseEntry;
  float sunVectorInBodyFrame[3];
} L1BGMI_S1_SUNDATA;

#undef _L1BGMI_S1_SUNDATA_
#endif


#ifndef _L1BGMI_S1_CALCOUNTS_
define _L1BGMI_S1_CALCOUNTS_

typedef struct {
  float hotLoadThermisterTemp[9][11];
  unsigned short hotLoadReading[9][65];
  unsigned short coldLoadReading[9][85];
  unsigned short hotLoadnDiodeReading[9][65];
  unsigned short coldLoadnDiodeReading[9][85];
} L1BGMI_S1_CALCOUNTS;

#undef _L1BGMI_S1_CALCOUNTS_
#endif

#ifndef _L1BGMI_S1_CALIBRATION_
define _L1BGMI_S1_CALIBRATION_

typedef struct {
  float hotLoadTemp[9];
  float coldSkyTemp[9];
  float onOrbitNonLinearity[9];
  float derivedNonLinearity[9];
  unsigned short meanHotLoadCount[9];
  unsigned short meanHotLoadCntnDiode[9];
  unsigned short meanColdSkyCount[9];
  unsigned short meanColdSkyCntnDiode[9];
  float diodeCoupledTemp[9];
} L1BGMI_S1_CALIBRATION;

#undef _L1BGMI_S1_CALIBRATION_
#endif
float gain[9][2];
float offset[9][2];
float nonLinearGain[9];
short calibrationQCflag;
short diodeFlag;
float receiverTemp[9];
float receiverGain[9];
} L1BGMI_S1_CALIBRATION;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;
#endif

#ifndef _L1BGMI_S1_SAMPLEHEADER_
#define _L1BGMI_S1_SAMPLEHEADER_

typedef struct {
    signed char blanking;
    short earthViewFirstSample;
    short sampleNumber[9][4];
    unsigned int tachSeconds[32];
    unsigned short tachMicroSeconds[32];
typedef struct {
    signed char dataQuality;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    double FractionalGranuleNumber;
} L1BGMI_S1_SCANSTATUS;

#endif

#include "L1BGMI_S1.h"

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#include "L1BGMI_S1.h"
typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    L1BGMI_S1_SCANSTATUS scanStatus;
    L1BGMI_S1_SAMPLEHEADER sampleHeader;
    NAVIGATION navigation;
    L1BGMI_S1_CALIBRATION calibration;
    float moonVectorInstFrame[3];
    L1BGMI_S1_CALCOUNTS calCounts;
    L1BGMI_S1_SUNDATA sunData;
    float incidenceAngle[221];
    float satAzimuthAngle[221];
    float solarZenAngle[221];
    float solarAzimuthAngle[221];
    float sunGlintAngle[221];
    float Tb[221][9];
    short RFIFlag[221][5];
} L1BGMI_S1;

#endif

#ifndef _L1BGMI_SWATHS_
#define _L1BGMI_SWATHS_

typedef struct {
    L1BGMI_S1 S1;
    L1BGMI_S2 S2;
} L1BGMI_SWATHS;

#endif

#endif

**Fortran Structure Header file:**

```fortran
STRUCTURE /L1BGMI_S2_SUNDATA/
    REAL*4 solarBetaAngle
    REAL*4 phaseFromOrbitMidnight
    REAL*4 sunEarthSeparation
    REAL*4 earthAngularRadius
    REAL*4 phaseOfEclipseExit
    REAL*4 orbitRate
```
REAL*4 timeSinceEclipseEntry
REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BGMI_S2_CALCOUNTS/
    REAL*4 hotLoadThermisterTemp(11,4)
    INTEGER*2 hotLoadReading(65,4)
    INTEGER*2 coldLoadReading(85,4)
    INTEGER*2 hotLoadnDiodeReading(65,4)
    INTEGER*2 coldLoadnDiodeReading(85,4)
END STRUCTURE

STRUCTURE /L1BGMI_S2_CALIBRATION/
    REAL*4 hotLoadTemp(4)
    REAL*4 coldSkyTemp(4)
    REAL*4 onOrbitNonLinearity(4)
    REAL*4 derivedNonLinearity(4)
    INTEGER*2 meanHotLoadCount(4)
    INTEGER*2 meanHotLoadCntnDiode(4)
    INTEGER*2 meanColdSkyCount(4)
    INTEGER*2 meanColdSkyCntnDiode(4)
    REAL*4 diodeCoupledTemp(4)
    REAL*4 gain(2,4)
    REAL*4 offset(2,4)
    REAL*4 nonLinearGain(4)
    INTEGER*2 calibrationQCflag
    INTEGER*2 diodeFlag
    REAL*4 receiverTemp(4)
    REAL*4 receiverGain(4)
END STRUCTURE

STRUCTURE /L1BGMI_S2_SAMPLEHEADER/
    BYTE blanking
    INTEGER*2 earthViewFirstSample
    INTEGER*2 sampleNumber(4,4)
    INTEGER*4 tachSeconds(32)
    INTEGER*2 tachMicroSeconds(32)
END STRUCTURE

STRUCTURE /L1BGMI_S2_SCANSTATUS/
    BYTE dataQuality
    BYTE missing
    BYTE modeStatus
5.18 1BGMI - GMI Brightness Temperatures

INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /L1BGMI_S2/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(221)
  REAL*4 Longitude(221)
  RECORD /L1BGMI_S2_SCANSTATUS/ scanStatus
  RECORD /L1BGMI_S2_SAMPLEHEADER/ sampleHeader
  RECORD /NAVIGATION/ navigation
  RECORD /L1BGMI_S2_CALIBRATION/ calibration
  REAL*4 moonVectorInstFrame(3)
  RECORD /L1BGMI_S2_CALCOUNTS/ calCounts
  RECORD /L1BGMI_S2_SUNDATA/ sunData
  REAL*4 incidenceAngle(221)
  REAL*4 satAzimuthAngle(221)
  REAL*4 solarZenAngle(221)
  REAL*4 solarAzimuthAngle(221)
  REAL*4 sunGlintAngle(221)
  REAL*4 Tb(4,221)
  INTEGER*2 RFIFlag(2,221)
END STRUCTURE

STRUCTURE /L1BGMI_S1_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
  REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BGMI_S1_CALCOUNTS/
  REAL*4 hotLoadThermisterTemp(11,9)
INTEGER*2 hotLoadReading(65,9)
INTEGER*2 coldLoadReading(85,9)
INTEGER*2 hotLoadnDiodeReading(65,9)
INTEGER*2 coldLoadnDiodeReading(85,9)
END STRUCTURE

STRUCTURE /L1BGMI_S1_CALIBRATION/
  REAL*4 hotLoadTemp(9)
  REAL*4 coldSkyTemp(9)
  REAL*4 onOrbitNonLinearity(9)
  REAL*4 derivedNonLinearity(9)
  INTEGER*2 meanHotLoadCount(9)
  INTEGER*2 meanHotLoadCntnDiode(9)
  INTEGER*2 meanColdSkyCount(9)
  INTEGER*2 meanColdSkyCntnDiode(9)
  REAL*4 diodeCoupledTemp(9)
  REAL*4 gain(2,9)
  REAL*4 offset(2,9)
  REAL*4 nonLinearGain(9)
  INTEGER*2 calibrationQCflag
  INTEGER*2 diodeFlag
  REAL*4 receiverTemp(9)
  REAL*4 receiverGain(9)
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
  REAL*4 scAlt
  REAL*4 dprAlt
  REAL*4 scAttRollGeoc
  REAL*4 scAttPitchGeoc
  REAL*4 scAttYawGeoc
  REAL*4 scAttRollGeod
  REAL*4 scAttPitchGeod
  REAL*4 scAttYawGeod
  REAL*4 greenHourAng
  REAL*8 timeMidScan
  REAL*8 timeMidScanOffset
END STRUCTURE
STRUCTURE /L1BGMI_S1_SAMPLEHEADER/
  BYTE blanking
  INTEGER*2 earthViewFirstSample
  INTEGER*2 sampleNumber(4,9)
  INTEGER*4 tachSeconds(32)
  INTEGER*2 tachMicroSeconds(32)
END STRUCTURE

STRUCTURE /L1BGMI_S1_SCANSTATUS/
  BYTE dataQuality
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BGMI_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(221)
  REAL*4 Longitude(221)
  RECORD /L1BGMI_S1_SCANSTATUS/ scanStatus
  RECORD /L1BGMI_S1_SAMPLEHEADER/ sampleHeader
  RECORD /NAVIGATION/ navigation
  RECORD /L1BGMI_S1_CALIBRATION/ calibration
  REAL*4 moonVectorInstFrame(3)
5.19 1BTMI - TMI unpacked packet data

1BTMI contains TMI science data from the TMI passive microwave instrument flown on the TRMM satellite. There are 3 swaths. Swath S1 has 10V 10H; Swath S2 has 19V, 19H, 21V, 37V, 37H; Swath S3 has 85V, 85H;

The S1 channels are:
10.7 GHz vertically-polarized
10.7 GHz horizontally-polarized

The S2 channels are:
18.7 GHz vertically-polarized
18.7 GHz horizontally-polarized
23.8 GHz vertically-polarized
36.5 GHz vertically-polarized
36.5 GHz horizontally-polarized

The S3 channels are:
85.0 GHz vertically-polarized
85.0 GHz horizontally-polarized

Earth observations are taken during a segment of the rotation when TMI is looking in the +x direction of the TRMM satellite. Since the spacecraft turns around every few weeks,
+x may be forward or aft. We define the spacecraft axis v, used in the definition of the variable SCorientation, at the center of this segment and the same as the +x direction.

Before Aug 7, 2001 $31.6 \text{rpm} \times \frac{1 \text{min}}{60 \text{s}} \times \frac{5490 \text{s}}{\text{orbit}} = 2891 \text{ scans / orbit}.$

After Aug 24, 2001 $31.6 \text{rpm} \times \frac{1 \text{min}}{60 \text{s}} \times \frac{5550 \text{s}}{\text{orbit}} = 2923 \text{ scans / orbit}.$

RELATION BETWEEN THE SWATHS: Swath S2 has the same number of scans and the same number of pixels as Swath S1. Swath S3 has the same number of scans and twice as many pixels as Swath S1. Each S1 scan contains 2 channels sampled 104 times along the scan. Each S2 scan contains 5 channels sampled 104 times along the scan. Each S3 scan contains 2 channels sampled 208 times along the scan.

Dimension definitions:
VH 2 Number of polarizations.
nscan1 var Typical number of Swath S1 scans in the granule.
nchannel1 2 Number of Swath S1 channels (10V).
nfreq1 1 Number of frequencies in Swath 1.
npixelevl1 104 Number of earth view pixels in one scan.
npixelhtl1 8 Number of hot load pixels in one scan.
npixelcs1 8 Number of cold sky pixels in one scan.
nscan2 var Typical number of Swath S2 scans in the granule.
nchannel2 5 Number of Swath S2 channels (19V 19H 21V 37V 37H).
nfreq2 3 Number of frequencies in Swath 2.
npixelevl2 104 Number of earth view pixels in one scan.
npixelhtl2 8 Number of hot load pixels in one scan.
npixelcs2 8 Number of cold sky pixels in one scan.
nscan3 var Typical number of Swath S3 scans in the granule.
nchannel3 2 Number of Swath S3 channels (85V 85H).
nfreq3 1 Number of frequencies in Swath 3.
npixelevl3 208 Number of earth view pixels in one scan.
npixelhtl3 16 Number of hot load pixels in one scan.
npixelcs3 16 Number of cold sky pixels in one scan.
nchannelall 9 Number of all channels.
ntherm 3 Number of hot load thermisters.
LNL 2 Linear and non-linear.
nndiode 6 Number of noise diodes.
dim2 2 Number.
dim3 3 Number.
dim4 4 Number.
dim5 5 Number.
dim6 6 Number.
dim7 7 Number.
dim8 8 Number.
dim9 9 Number.
dim10 10 Number.
dim11 11 Number.
dim12 12 Number.
TMIxyz 3 x, y, z components in TMI instrument coordinate system.

Figure 287 through Figure 308 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 287: Data Format Structure for 1BTMI, TMI unpacked packet data

Figure 288: Data Format Structure for 1BTMI, S1
Figure 289: Data Format Structure for 1BTMI, S2
5.19 1BTMI - TMI unpacked packet data

Figure 290: Data Format Structure for 1BTMI, S3

Figure 291: Data Format Structure for 1BTMI, S1, ScanTime
Figure 292: Data Format Structure for 1BTMI, S1, scanStatus
5.19 1BTMI - TMI unpacked packet data

Figure 293: Data Format Structure for 1BTMI, S1, navigation

Figure 294: Data Format Structure for 1BTMI, S1, calibration
Figure 295: Data Format Structure for 1BTMI, S1, calCounts

- hotLoadThermisterTemp 4 bytes
  Array: ntherm x nchannel1 x nscan1
- hotLoadReading 2 bytes
  Array: npixelht1 x nchannel1 x nscan1
- coldLoadReading 2 bytes
  Array: npixelcs1 x nchannel1 x nscan1

Figure 296: Data Format Structure for 1BTMI, S1, sunData

- solarBetaAngle 4 bytes
  Array: nsan1
- phaseFromOrbitMidnight 4 bytes
  Array: nsan1
- sunEarthSeparation 4 bytes
  Array: nsan1
- earthAngularRadius 4 bytes
  Array: nsan1
- phaseOfEclipseExit 4 bytes
  Array: nsan1
- orbitRate 4 bytes
  Array: nsan1
- timeSinceEclipseEntry 4 bytes
  Array: nsan1
- sunVectorInBodyFrame 4 bytes
  Array: 3 x nsan1

Figure 297: Data Format Structure for 1BTMI, S2, ScanTime

- Year 2 bytes
  Array: nsan2
- Month 1 byte
  Array: nsan2
- DayOfMonth 1 byte
  Array: nsan2
- Hour 1 byte
  Array: nsan2
- Minute 1 byte
  Array: nsan2
- Second 1 byte
  Array: nsan2
- MilliSecond 2 bytes
  Array: nsan2
- DayOfYear 2 bytes
  Array: nsan2
- SecondOfDay 8 bytes
  Array: nsan2
5.19 1BTMI - TMI unpacked packet data

Figure 298: Data Format Structure for 1BTMI, S2, scanStatus
Figure 299: Data Format Structure for 1BTMI, S2, navigation
5.19 1BTMI - TMI unpacked packet data

- **hotLoadTemp**: 4 bytes, Array: nchannel2 x nscan2
- **coldSkyTemp**: 4 bytes, Array: nchannel2 x nscan2
- **onOrbitNonLinearity**: 4 bytes, Array: nchannel2 x nscan2
- **meanHotLoadCount**: 2 bytes, Array: nchannel2 x nscan2
- **meanColdSkyCount**: 2 bytes, Array: nchannel2 x nscan2
- **gain**: 4 bytes, Array: LNL x nchannel2 x nscan2
- **offset**: 4 bytes, Array: LNL x nchannel2 x nscan2
- **nonLinearGain**: 4 bytes, Array: nchannel2 x nscan2
- **calibrationQCflag**: 2 bytes, Array: nscan2
- **receiverTemp**: 4 bytes, Array: nchannel2 x nscan2
- **receiverGain**: 4 bytes, Array: nchannel2 x nscan2

**Figure 300: Data Format Structure for 1BTMI, S2, calibration**

- **hotLoadThermisterTemp**: 4 bytes, Array: ntherm x nchannel2 x nscan2
- **hotLoadReading**: 2 bytes, Array: npixelht2 x nchannel2 x nscan2
- **coldLoadReading**: 2 bytes, Array: npixelcs2 x nchannel2 x nscan2

**Figure 301: Data Format Structure for 1BTMI, S2, calCounts**

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo (Metadata):**
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.
Figure 302: Data Format Structure for 1BTMI, S2, sunData

Figure 303: Data Format Structure for 1BTMI, S3, ScanTime
5.19 1BTMI - TMI unpacked packet data

Figure 304: Data Format Structure for 1BTMI, S3, scanStatus
STANDARD GPM PRODUCTS

**navigation**
- `scPos`: 4 bytes, Array: XYZ x nscan3
- `scVel`: 4 bytes, Array: XYZ x nscan3
- `scLat`: 4 bytes, Array: nscan3
- `scLon`: 4 bytes, Array: nscan3
- `scAlt`: 4 bytes, Array: nscan3
- `dprAlt`: 4 bytes, Array: nscan3
- `scAttRollGeoc`: 4 bytes, Array: nscan3
- `scAttPitchGeoc`: 4 bytes, Array: nscan3
- `scAttYawGeoc`: 4 bytes, Array: nscan3
- `scAttRollGeod`: 4 bytes, Array: nscan3
- `scAttPitchGeod`: 4 bytes, Array: nscan3
- `scAttYawGeod`: 4 bytes, Array: nscan3
- `greenHourAng`: 4 bytes, Array: nscan3
- `timeMidScan`: 8 bytes, Array: nscan3
- `timeMidScanOffset`: 8 bytes, Array: nscan3

**Figure 305: Data Format Structure for 1BTMI, S3, navigation**

**calibration**
- `hotLoadTemp`: 4 bytes, Array: nchannel3 x nscan3
- `coldSkyTemp`: 4 bytes, Array: nchannel3 x nscan3
- `onOrbitNonLinearity`: 4 bytes, Array: nchannel3 x nscan3
- `meanHotLoadCount`: 2 bytes, Array: nchannel3 x nscan3
- `meanColdSkyCount`: 2 bytes, Array: nchannel3 x nscan3
- `gain`: 4 bytes, Array: LNL x nchannel3 x nscan3
- `offset`: 4 bytes, Array: LNL x nchannel3 x nscan3
- `nonLinearGain`: 4 bytes, Array: nchannel3 x nscan3
- `calibrationQClflag`: 2 bytes, Array: nscan3
- `receiverTemp`: 4 bytes, Array: nchannel3 x nscan3
- `receiverGain`: 4 bytes, Array: nchannel3 x nscan3

**Figure 306: Data Format Structure for 1BTMI, S3, calibration**
5.19 1BTMI - TMI unpacked packet data

**calCounts**
- **hotLoadThermisterTemp** 4 bytes
  - Array: ntherm x nchannel3 x nscan3
- **hotLoadReading** 2 bytes
  - Array: npixelht3 x nchannel3 x nscan3
- **coldLoadReading** 2 bytes
  - Array: npixelcs3 x nchannel3 x nscan3

Figure 307: Data Format Structure for 1BTMI, S3, calCounts

**sunData**
- **solarBetaAngle** 4 bytes
  - Array: nscan3
- **phaseFromOrbitMidnight** 4 bytes
  - Array: nscan3
- **sunEarthSeparation** 4 bytes
  - Array: nscan3
- **earthAngularRadius** 4 bytes
  - Array: nscan3
- **phaseOfEclipseExit** 4 bytes
  - Array: nscan3
- **orbitRate** 4 bytes
  - Array: nscan3
- **timeSinceEclipseEntry** 4 bytes
  - Array: nscan3
- **sunVectorInBodyFrame** 4 bytes
  - Array: 3 x nscan3

Figure 308: Data Format Structure for 1BTMI, S3, sunData

**S1 (Swath)**

**S1_SwathHeader** (Metadata): SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in S1)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- -9999 Missing value

**Month** (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
- -99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
- -99 Missing value
**Hour** (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npixelx1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixelx1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S1)

**dataQuality** (1-byte char, array size: nscan1):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
</tbody>
</table>
modeStatus is not normal

QAC errors associated with this scan

missing (1-byte char, array size: nscan1):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

modeStatus (1-byte char, array size: nscan1):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SC orientation is not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine tmIsStatus</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nscan1):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

**Bit Meaning if bit = 1**
- 0 Latitude limit exceeded for viewed pixel locations
- 1 Negative scan time, invalid input
- 2 Error getting spacecraft attitude at scan mid-time
- 3 Error getting spacecraft ephemeris at scan mid-time
- 4 Invalid input non-unit ray vector for any pixel
- 5 Ray misses Earth for any pixel with normal pointing
- 6 Nadir calculation error for subsatellite position
- 7 Pixel count with geolocation error over threshold
- 8 Error in getting spacecraft attitude for any pixel
- 9 Error in getting spacecraft ephemeris for any pixel
- 10 Spare (always 0)
- 11 Spare (always 0)
- 12 Spare (always 0)
- 13 Spare (always 0)
- 14 Spare (always 0)
- 15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan1):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is \(2^{*i}\)):

**Bit Meaning if bit = 1**
- 0 Ephemeris Gap Interpolated
- 1 Attitude Gap Interpolated
- 2 Attitude jump/discontinuity
- 3 Attitude out of range
- 4 Anomalous Time Step
- 5 GHA not calculated due to error
- 6 SunData (Group) not calculated due to error
- 7 Failure to calculate Sun in inertial coordinates
- 8 Fallback to GES ephemeris
- 9 Fallback to GEONS ephemeris
- 10 Fallback to PVT ephemeris
- 11 Fallback to OBP ephemeris
- 12 Spare (always 0)
- 13 Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

SCorientation (2-byte integer, array size: nscan1):
The positive angle of the spacecraft vector \(v\) from the satellite forward direction of motion, measured clockwise facing down. We define \(v\) in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8002</td>
<td>Yaw turn in progress</td>
</tr>
<tr>
<td>-8003</td>
<td>Deep Space Calibration in progress</td>
</tr>
<tr>
<td>-8004</td>
<td>Non-nominal pointing other than above</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan1):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal ACS mode (4) for mission science</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal ACS mode</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan1):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>
targetSelectionMidScan (1-byte integer, array size: nscan1):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control
System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

tmiIsStatus (1-byte char, array size: nscan1):
Status of the instrument from Housekeeping packets. Bit 0 is the most significant bit
(I.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**(8-i) - 1).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>01</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>02</td>
<td>Spare command 1 Status</td>
</tr>
<tr>
<td>03</td>
<td>Spare command 2 Status</td>
</tr>
<tr>
<td>04</td>
<td>1 Hz Clock Select (1=A, 0=B)</td>
</tr>
<tr>
<td>05</td>
<td>Spare</td>
</tr>
<tr>
<td>06</td>
<td>Spare Command 4 Status</td>
</tr>
<tr>
<td>07</td>
<td>Spare Command 5 Status</td>
</tr>
</tbody>
</table>

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

attDetermSource (2-byte integer, array size: nscan1):

Attitude determination source.
A flag explaining how the attitude value was calculated.
Improved estimates make use of ground processing of PR
science-instrument-measured roll values, Gyroscope data,
and Sun Sensor 1 data. Earlier products (TRMM V7 and before)
used the onboard attitudes with various corrections.
Values were determined for each granule based on the data
available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
</tbody>
</table>
421 Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)
413 Reduced accuracy, sun data not available (affecting pitch)
411 Reduced accuracy, PR roll and sun sensor not available
300-399 Reduced accuracy due to various special conditions
200-299 Fallback to using the onboard attitude estimates with TRMM V7 corrections
-91 Spacecraft in safehold mode, no science data
-99 No data due to telemetry data gap

TRMMyawUpdateS (1-byte integer, array size: nscan1):

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor
for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMqac** (1-byte integer, array size: nscan1):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

**navigation** (Group in S1)

**scPos** (4-byte float, array size: XYZ x nscan1):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

**scVel** (4-byte float, array size: XYZ x nscan1):
The velocity vector (ms\(^{-1}\)) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

**scLat** (4-byte float, array size: nscan1):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

**scLon** (4-byte float, array size: nscan1):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAlt** (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value
dprAlt (4-byte float, array size: nscan1):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeoc (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeoc (4-byte float, array size: nscan1):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeod (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

\textbf{scAttYawGeo} (4-byte float, array size: nscan1):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

\textbf{greenHourAng} (4-byte float, array size: nscan1):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

\textbf{timeMidScan} (8-byte float, array size: nscan1):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

\textbf{timeMidScanOffset} (8-byte float, array size: nscan1):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

\textbf{calibration} (Group in S1)

\textbf{hotLoadTemp} (4-byte float, array size: nchannel1 x nscan1):
The mean physical temperature for the temperature sensors attached to the hot load. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

\textbf{coldSkyTemp} (4-byte float, array size: nchannel1 x nscan1):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

\textbf{onOrbitNonLinearity} (4-byte float, array size: nchannel1 x nscan1):
The on Orbit Non-Linearity. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

\textbf{meanHotLoadCount} (2-byte unsigned integer, array size: nchannel1 x nscan1):
The mean Hot Load Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value

\textbf{meanColdSkyCount} (2-byte unsigned integer, array size: nchannel1 x nscan1):
The mean Cold Sky Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value
gain (4-byte float, array size: LNL x nchannel1 x nscan1):
Automatic gain control. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

offset (4-byte float, array size: LNL x nchannel1 x nscan1):
Offset. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

nonLinearGain (4-byte float, array size: nchannel1 x nscan1):
The nonlinear gain. Special values are defined as:
-9999.9 Missing value

calibrationQCflag (2-byte integer, array size: nscan1):
calibrationQCflag. Values range from 0 to 15. Special values are defined as:
-9999 Missing value

receiverTemp (4-byte float, array size: nchannel1 x nscan1):
The receiver temperature. Special values are defined as:
-9999.9 Missing value

receiverGain (4-byte float, array size: nchannel1 x nscan1):
The receiver gain. Special values are defined as:
-9999.9 Missing value

moonVectorInstFrame (4-byte float, array size: TMIxyz x nscan1):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
-9999.9 Missing value

calCounts (Group in S1)

hotLoadThermisterTemp (4-byte float, array size: ntherm x nchannel1 x nscan1):
Hot Load Thermister Temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

hotLoadReading (2-byte unsigned integer, array size: npixelht1 x nchannel1 x nscan1):
Hot Load Reading. Values range from 0 to 15 counts. Special values are defined as:
0 Missing value

coldLoadReading (2-byte unsigned integer, array size: npixelscs1 x nchannel1 x nscan1):
Cold Load Reading. Values range from 0 to 15 counts. Special values are defined as:
0 Missing value

sunData (Group in S1)
solarBetaAngle (4-byte float, array size: nscan1):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseFromOrbitMidnight (4-byte float, array size: nscan1):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

sunEarthSeparation (4-byte float, array size: nscan1):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

earthAngularRadius (4-byte float, array size: nscan1):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

phaseOfEclipseExit (4-byte float, array size: nscan1):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

orbitRate (4-byte float, array size: nscan1):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

timeSinceEclipseEntry (4-byte float, array size: nscan1):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

sunVectorInBodyFrame (4-byte float, array size: 3 x nscan1):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value
incidenceAngle (4-byte float, array size: nchannel1 x npixelev1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

satAzimuthAngle (4-byte float, array size: npixelev1 x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarZenAngle (4-byte float, array size: npixelev1 x nscan1):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarAzimuthAngle (4-byte float, array size: npixelev1 x nscan1):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (4-byte float, array size: npixelev1 x nscan1):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

Tb (4-byte float, array size: nchannel1 x npixelev1 x nscan1):
Earth view brightness temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

RFIFlag (2-byte integer, array size: nfreq1 x npixelev1 x nscan1):
Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: No RFI on earth view samples and all Tb values of this swath are lower than or equal to 320 K.
1: Earth view Tb values from one or more channels of this swath are greater than 320 K.
2: RFI on earth view samples is detected by spectral differential method (10
GHz and 19 GHz channels only).
3: (combination of 1 and 2). Earth view Tb values from one or more channels of this swath are greater than 320 K and RFI is detected by spectral differential method (10 GHz and 19 GHz channels only)
-9999: Missing

**S2 (Swath)**

**S2_SwathHeader (Metadata):**
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime (Group in S2)**
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-99999 Missing value
DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: npixelev2 x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: npixelev2 x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scanStatus (Group in S2)

dataQuality (1-byte char, array size: nscan2):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not normal</td>
</tr>
<tr>
<td>7</td>
<td>QAC errors associated with this scan</td>
</tr>
</tbody>
</table>

missing (1-byte char, array size: nscan2):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
</tbody>
</table>
modeStatus (1-byte char, array size: nscan2):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if
the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other
bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation is not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine tmiIsStatus</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nscan2):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
</tbody>
</table>
5.19 1BTMI - TMI unpacked packet data

8  Error in getting spacecraft attitude for any pixel
9  Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan2):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan2):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
</tbody>
</table>
180  -X forward (yaw 180)
-8002 Yaw turn in progress
-8003 Deep Space Calibration in progress
-8004 Non-nominal pointing other than above
-9999 Missing

**pointingStatus** (2-byte integer, array size: nscan2):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal ACS mode (4) for mission science</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal ACS mode</td>
</tr>
</tbody>
</table>

**acsModeMidScan** (1-byte integer, array size: nscan2):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: nscan2):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>
**tmiIsStatus** (1-byte char, array size: nscan2):
Status of the instrument from Housekeeping packets. Bit 0 is the most significant bit (i.e., if bit \(i = 1\) and other bits = 0, the unsigned integer value is \(2^{**(8-i) - 1}**\).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Receiver status (0=\text{ON}, 1=\text{OFF})</td>
</tr>
<tr>
<td>01</td>
<td>Spinup Status (0=\text{ON}, 1=\text{OFF})</td>
</tr>
<tr>
<td>02</td>
<td>Spare command 1 Status</td>
</tr>
<tr>
<td>03</td>
<td>Spare command 2 Status</td>
</tr>
<tr>
<td>04</td>
<td>1 Hz Clock Select (i=\text{A}, 0=\text{B})</td>
</tr>
<tr>
<td>05</td>
<td>Spare</td>
</tr>
<tr>
<td>06</td>
<td>Spare Command 4 Status</td>
</tr>
<tr>
<td>07</td>
<td>Spare Command 5 Status</td>
</tr>
</tbody>
</table>

**FractionalGranuleNumber** (8-byte float, array size: nscan2):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9   Missing value

**attDetermSource** (2-byte integer, array size: nscan2):

Attitude determination source.
A flag explaining how the attitude value was calculated.
Improved estimates make use of ground processing of PR science-instrument-measured roll values, Gyroscope data, and Sun Sensor 1 data. Earlier products (TRMM V7 and before) used the onboard attitudes with various corrections. Values were determined for each granule based on the data available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430 and higher</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
<tr>
<td>421</td>
<td>Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)</td>
</tr>
<tr>
<td>413</td>
<td>Reduced accuracy, sun data not available (affecting pitch)</td>
</tr>
<tr>
<td>411</td>
<td>Reduced accuracy, PR roll and sun sensor not available</td>
</tr>
<tr>
<td>300-399</td>
<td>Reduced accuracy due to various special conditions</td>
</tr>
<tr>
<td>200-299</td>
<td>Fallback to using the onboard attitude estimates with TRMM V7 corrections</td>
</tr>
<tr>
<td>-91</td>
<td>Spacecraft in safehold mode, no science data</td>
</tr>
<tr>
<td>-99</td>
<td>No data due to telemetry data gap</td>
</tr>
</tbody>
</table>
TRMMcontMode (1-byte integer, array size: nscan2):

The Contingency Mode Flag from telemetry indicates alternate attitude control of the spacecraft. The nominal at-launch Attitude Control System (ACS) for TRMM used Earth horizon sensors for pitch and roll control, and the yaw was updated twice each orbit using the Sun Sensors and propagated using gyro data. However, due to possible problems identified with the Earth Sensor Assembly (ESA) lifetime on-orbit, a contingency ACS mode was developed late in the development cycle. This mode used the Sun Sensors, magnetometers, and gyroscope data. It proved very valuable when the horizon sensors had problems with TRMM moving to the higher operating altitude (from 350 to 402.5 km) to extend the mission lifetime. Thus the contingency mode was used throughout the post-boost period. It was also tested early in the mission on 1998-01-13.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal control of spacecraft used in the pre-boost period</td>
</tr>
<tr>
<td>1</td>
<td>Contingency mode control used in the post-boost period</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

TRMMyawUpdateS (1-byte integer, array size: nscan2):

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>
TRMMpqac (1-byte integer, array size: nscan2):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

**navigation** (Group in S2)

**scPos** (4-byte float, array size: XYZ x nscan2):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9  Missing value

**scVel** (4-byte float, array size: XYZ x nscan2):
The velocity vector ($ms^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9  Missing value

**scLat** (4-byte float, array size: nscan2):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9  Missing value

**scLon** (4-byte float, array size: nscan2):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAlt** (4-byte float, array size: nscan2):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**dprAlt** (4-byte float, array size: nscan2):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**scAttRollGeoc** (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the
Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan2):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan2):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan2):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value
timeMidScan (8-byte float, array size: nscan2):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980.
timeMidScan is used as the reference time for the scPos and scVel values. Values range
from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan2):
Offset from the secondary header packet time to the timeMidScan. Values range from 0
to 100 s. Special values are defined as:
-9999.9 Missing value

calibration (Group in S2)

hotLoadTemp (4-byte float, array size: nchannel2 x nscan2):
The mean physical temperature for the temperature sensors attached to the hot load.
Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

coldSkyTemp (4-byte float, array size: nchannel2 x nscan2):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined
as:
-9999.9 Missing value

onOrbitNonLinearity (4-byte float, array size: nchannel2 x nscan2):
The on Orbit Non-Linearity. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

meanHotLoadCount (2-byte unsigned integer, array size: nchannel2 x nscan2):
The mean Hot Load Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value

meanColdSkyCount (2-byte unsigned integer, array size: nchannel2 x nscan2):
The mean Cold Sky Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value

gain (4-byte float, array size: LNL x nchannel2 x nscan2):
Automatic gain control. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

offset (4-byte float, array size: LNL x nchannel2 x nscan2):
Offset. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

nonLinearGain (4-byte float, array size: nchannel2 x nscan2):
The nonlinear gain. Special values are defined as:
-9999.9 Missing value
**calibrationQCflag** (2-byte integer, array size: nscan2):  
*calibrationQCflag*. Values range from 0 to 15. Special values are defined as:  
-9999 Missing value

**receiverTemp** (4-byte float, array size: nchannel2 x nscan2):  
The receiver temperature. Special values are defined as:  
-9999.9 Missing value

**receiverGain** (4-byte float, array size: nchannel2 x nscan2):  
The receiver gain. Special values are defined as:  
-9999.9 Missing value

**moonVectorInstFrame** (4-byte float, array size: TMIxyz x nscan2):  
The x, y, z components of the moon vector in the GMI instrument coordinate system.  
Values are in counts. Special values are defined as:  
-9999.9 Missing value

**calCounts** (Group in S2)

**hotLoadThermisterTemp** (4-byte float, array size: ntherm x nchannel2 x nscan2):  
Hot Load Thermister Temperature. Values range from 0 to 400 K. Special values are defined as:  
-9999.9 Missing value

**hotLoadReading** (2-byte unsigned integer, array size: npixelht2 x nchannel2 x nscan2):  
Hot Load Reading. Values range from 0 to 15 counts. Special values are defined as:  
0 Missing value

**coldLoadReading** (2-byte unsigned integer, array size: npixelcs2 x nchannel2 x nscan2):  
Cold Load Reading. Values range from 0 to 15 counts. Special values are defined as:  
0 Missing value

**sunData** (Group in S2)

**solarBetaAngle** (4-byte float, array size: nscan2):  
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:  
-9999.9 Missing value

**phaseFromOrbitMidnight** (4-byte float, array size: nscan2):  
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees
occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:

-9999.9 Missing value

**sunEarthSeparation** (4-byte float, array size: nscan2):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:

-9999.9 Missing value

**earthAngularRadius** (4-byte float, array size: nscan2):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:

-9999.9 Missing value

**phaseOfEclipseExit** (4-byte float, array size: nscan2):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:

-9999.9 Missing value

**orbitRate** (4-byte float, array size: nscan2):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:

-9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan2):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:

-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan2):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:

-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npixelev2 x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:

-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npixelev2 x nscan2):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value
solarZenAngle (4-byte float, array size: npixelev2 x nscan2):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

solarAzimuthAngle (4-byte float, array size: npixelev2 x nscan2):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (4-byte float, array size: npixelev2 x nscan2):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

Tb (4-byte float, array size: nchannel2 x npixelev2 x nscan2):
Earth view brightness temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

RFIFlag (2-byte integer, array size: nfreq1 x npixelev2 x nscan2):
Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:
0: No RFI on earth view samples and all Tb values of this swath are lower than or equal to 320 K.
1: Earth view Tb values from one or more channels of this swath are greater than 320 K.
2: RFI on earth view samples is detected by spectral differential method (10 GHz and 19 GHz channels only).
3: (combination of 1 and 2). Earth view Tb values from one or more channels of this swath are greater than 320 K and RFI is detected by spectral differential method (10 GHz and 19 GHz channels only).
-9999: Missing
S3 (Swath)

S3_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in S3)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan3):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan3):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan3):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan3):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan3):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan3):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan3):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan3):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan3):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixlev3 x nscan3):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixelev3 x nsan3):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in S3)

**dataQuality** (1-byte char, array size: nsan3):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not normal</td>
</tr>
<tr>
<td>7</td>
<td>QAC errors associated with this scan</td>
</tr>
</tbody>
</table>

**missing** (1-byte char, array size: nsan3):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte char, array size: nsan3):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit =
1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation is not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine tmiIsStatus</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan3):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0 the unsigned integer value is \( 2^i \)).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
**geoWarning** (2-byte integer, array size: nscan3):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^{**i}$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan3):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8002</td>
<td>Yaw turn in progress</td>
</tr>
<tr>
<td>-8003</td>
<td>Deep Space Calibration in progress</td>
</tr>
<tr>
<td>-8004</td>
<td>Non-nominal pointing other than above</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan3):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.
### Value Meaning

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal ACS mode (4) for mission science</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal ACS mode</td>
</tr>
</tbody>
</table>

### `acsModeMidScan` (1-byte integer, array size: nscan3):

`acsModeMidScan` is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

### `targetSelectionMidScan` (1-byte integer, array size: nscan3):

`targetSelectionMidScan` is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

### `tmiIsStatus` (1-byte char, array size: nscan3):

Status of the instrument from Housekeeping packets. Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is \(2^{*\text{8-i}} - 1\)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Receiver status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>01</td>
<td>Spinup Status (0=ON, 1=OFF)</td>
</tr>
<tr>
<td>02</td>
<td>Spare command 1 Status</td>
</tr>
<tr>
<td>03</td>
<td>Spare command 2 Status</td>
</tr>
<tr>
<td>04</td>
<td>1 Hz Clock Select (1=A, 0=B)</td>
</tr>
<tr>
<td>05</td>
<td>Spare</td>
</tr>
</tbody>
</table>
FractionalGranuleNumber (8-byte float, array size: nscan3):
The floating point granule number. The granule begins at the Southern-most point of 
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the 
spacecraft is halfway through granule 10 and starting the descending half of the granule. 
Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

attDetermSource (2-byte integer, array size: nscan3):

Attitude determination source. 
A flag explaining how the attitude value was calculated. 
Improved estimates make use of ground processing of PR 
science-instrument-measured roll values, Gyroscope data, 
and Sun Sensor 1 data. Earlier products (TRMM V7 and before) 
used the onboard attitudes with various corrections. 
Values were determined for each granule based on the data 
available and conditions for each orbit. Flag values follow.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>430 and higher</td>
<td>Best accuracy, good data for this orbit</td>
</tr>
<tr>
<td>421</td>
<td>Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)</td>
</tr>
<tr>
<td>413</td>
<td>Reduced accuracy, sun data not available (affecting pitch)</td>
</tr>
<tr>
<td>411</td>
<td>Reduced accuracy, PR roll and sun sensor not available</td>
</tr>
<tr>
<td>300-399</td>
<td>Reduced accuracy due to various special conditions</td>
</tr>
<tr>
<td>200-299</td>
<td>Fallback to using the onboard attitude estimates with TRMM V7 corrections</td>
</tr>
<tr>
<td>-91</td>
<td>Spacecraft in safehold mode, no science data</td>
</tr>
<tr>
<td>-99</td>
<td>No data due to telemetry data gap</td>
</tr>
</tbody>
</table>

TRMMcontMode (1-byte integer, array size: nscan3):

The Contingency Mode Flag from telemetry indicates 
alternate attitude control of the spacecraft. 
The nominal at-launch Attitude Control System (ACS) 
for TRMM used Earth horizon sensors for pitch and 
roll control, and the yaw was updated twice each orbit 
using the Sun Sensors and propagated using gyro data. 
However, due to possible problems identified with 
the Earth Sensor Assembly (ESA) lifetime on-orbit,
a contingency ACS mode was developed late in the
development cycle. This mode used the Sun Sensors,
magnetometers, and gyroscope data. It proved very
valuable when the horizon sensors had problems with
TRMM moving to the higher operating altitude (from
350 to 402.5 km) to extend the mission lifetime.
Thus the contingency mode was used throughout the
post-boost period. It was also tested early in

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal control of spacecraft used in the pre-boost period</td>
</tr>
<tr>
<td>1</td>
<td>Contingency mode control used in the post-boost period</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMYawUpdateS** (1-byte integer, array size: nscan3):

The Yaw Update Status flag in telemetry gives the status
of the Yaw accuracy for the nominal pre-boost Attitude Control
System (ACS) operation. The yaw is considered "indeterminate"
in various non-nominal control modes, and after the return
to the nominal Earth pointing (using the Earth sensor
for pitch and roll), the yaw is considered "inaccurate" until
the time when an "update" is done using a Sun sensor (at certain
positions in the orbit). Before the update "the yaw attitude
knowledge is acceptable for ACS use, but might not be acceptable
for science use" according to ACS Software User's Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRMMqac** (1-byte integer, array size: nscan3):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data.
If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this
format has a value of zero.

**navigation** (Group in S3)
scPos (4-byte float, array size: XYZ x nscan3):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coor-
dinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan3):
The velocity vector ($ms^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan3):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan3):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan3):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan3):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan3):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The or-
der of the components in the file is roll, pitch, and yaw. However, the angles are computed
using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll
for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital
Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the
Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft veloc-
ity opposite the orbit normal direction, and the X-axis is approximately in the velocity
direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that
pitch and roll will have twice orbital frequency components due to the onboard control
system following the oblate geodetic Earth horizon. Note also that the yaw value will
show an orbital frequency component relative to the Earth fixed ground track due to the
Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees.
Special values are defined as:
-9999.9 Missing value
5.19  1BTMI - TMI unpacked packet data

**scAttPitchGeoc** (4-byte float, array size: nscan3):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttYawGeoc** (4-byte float, array size: nscan3):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9  Missing value

**scAttRollGeod** (4-byte float, array size: nscan3):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttPitchGeod** (4-byte float, array size: nscan3):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttYawGeod** (4-byte float, array size: nscan3):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9  Missing value

**greenHourAng** (4-byte float, array size: nscan3):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9  Missing value

**timeMidScan** (8-byte float, array size: nscan3):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9  Missing value

**timeMidScanOffset** (8-byte float, array size: nscan3):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9  Missing value
calibration (Group in S3)

hotLoadTemp (4-byte float, array size: nchannel3 x nscan3):
The mean physical temperature for the temperature sensors attached to the hot load. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

coldSkyTemp (4-byte float, array size: nchannel3 x nscan3):
The mean cold sky temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

onOrbitNonLinearity (4-byte float, array size: nchannel3 x nscan3):
The on Orbit Non-Linearity. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

meanHotLoadCount (2-byte unsigned integer, array size: nchannel3 x nscan3):
The mean Hot Load Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value

meanColdSkyCount (2-byte unsigned integer, array size: nchannel3 x nscan3):
The mean Cold Sky Count. Values range from 0 to 15. Special values are defined as:
65535 Missing value

gain (4-byte float, array size: LNL x nchannel3 x nscan3):
Automatic gain control. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

offset (4-byte float, array size: LNL x nchannel3 x nscan3):
Offset. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

nonLinearGain (4-byte float, array size: nchannel3 x nscan3):
The nonlinear gain. Special values are defined as:
-9999.9 Missing value

calibrationQCflag (2-byte integer, array size: nscan3):
calibrationQCflag. Values range from 0 to 15. Special values are defined as:
-9999 Missing value

receiverTemp (4-byte float, array size: nchannel3 x nscan3):
The receiver temperature. Special values are defined as:
-9999.9 Missing value

receiverGain (4-byte float, array size: nchannel3 x nscan3):
The receiver gain. Special values are defined as:
-9999.9 Missing value
**moonVectorInstFrame** (4-byte float, array size: TMIxyz x nscan3):
The x, y, z components of the moon vector in the GMI instrument coordinate system. Values are in counts. Special values are defined as:
-9999.9 Missing value

**calCounts** (Group in S3)

**hotLoadThermisterTemp** (4-byte float, array size: ntherm x nchannel3 x nscan3):
Hot Load Thermister Temperature. Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

**hotLoadReading** (2-byte unsigned integer, array size: npixelht3 x nchannel3 x nscan3):
Hot Load Reading. Values range from 0 to 15 counts. Special values are defined as:
0 Missing value

**coldLoadReading** (2-byte unsigned integer, array size: npixelcs3 x nchannel3 x nscan3):
Cold Load Reading. Values range from 0 to 15 counts. Special values are defined as:
0 Missing value

**sunData** (Group in S3)

**solarBetaAngle** (4-byte float, array size: nscan3):
Sun direction elevation from the orbit plane, positive toward orbit normal which is given by the cross product of the spacecraft position and velocity vectors. Values range from -89.0 to 89.0 degrees. Special values are defined as:
-9999.9 Missing value

**phaseFromOrbitMidnight** (4-byte float, array size: nscan3):
Phase angle of the Sun direction around the orbit plane, with zero phase in the direction of the Earth center from the spacecraft and positive toward the spacecraft velocity direction so the phase increases with time. Zero phase occurs at local orbit midnight, 90 degrees occurs with the spacecraft over the Earth’s dawn terminator, 180 degrees occurs at local orbit noon, and -90 degrees occurs with the spacecraft over the Earth’s dusk terminator. Values range from -180.0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value

**sunEarthSeparation** (4-byte float, array size: nscan3):
The separation angle between the Sun and Earth directions from the spacecraft. Values range from 0 to 180.0 degrees. Special values are defined as:
-9999.9 Missing value
**earthAngularRadius** (4-byte float, array size: nscan3):
The angle between the center of the Earth and the horizon edge. The sun is above the Earth horizon when the sunEarthSeparation is greater than the earthAngularRadius. Values range from 69.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**phaseOfEclipseExit** (4-byte float, array size: nscan3):
The estimated phaseFromOrbitMidnight where the spacecraft leaves the Earth shadow, based on the instantaneous solarBetaAngle and earthAngularRadius. Values range from 0.0 to 80.0 degrees. Special values are defined as:
-9999.9 Missing value

**orbitRate** (4-byte float, array size: nscan3):
The instantaneous angular rate of the spacecraft around the orbit. Values range from 0.064 to 0.07 degrees/s. Special values are defined as:
-9999.9 Missing value

**timeSinceEclipseEntry** (4-byte float, array size: nscan3):
The estimated duration in seconds since the last entry into the Earth’s shadow. Values range from 0 to 5600.0 s. Special values are defined as:
-9999.9 Missing value

**sunVectorInBodyFrame** (4-byte float, array size: 3 x nscan3):
The unit sun vector direction in the TMI instrument body coordinate frame, defined such that +Z is nominally toward the Earth and gives the instrument spin axis, and data is collected nominally centered about the +X direction. Values range from 0 to 1.0. Special values are defined as:
-9999.9 Missing value

**incidenceAngle** (4-byte float, array size: npixelev3 x nscan3):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npixelev3 x nscan3):
The angle clockwise looking down between the local pixel geodetic north and the direction to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: npixelev3 x nscan3):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: npixelev3 x nscan3):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
sunGlintAngle (4-byte float, array size: npixelev3 x nscan3):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. sunGlintAngle is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
- 9999.9  Missing value

Tb (4-byte float, array size: nchannel3 x npixelev3 x nscan3):
Earth view brightness temperature. Values range from 0 to 400 K. Special values are defined as:
- 9999.9  Missing value

RFIFlag (2-byte integer, array size: nfreq1 x npixelev3 x nscan3):

Radio Frequency Interference (RFI) Flag. The flag is set to non-zero if the pixel is contaminated by RFI according to certain filters. Current values are:

0: No RFI on earth view samples and all Tb values of this swath are lower than or equal to 320 K.
1: Earth view Tb values from one or more channels of this swath are greater than 320 K.
-9999: Missing

C Structure Header file:

```c
#ifndef _TK_1BTMI_H_
define _TK_1BTMI_H_
#endif

#ifndef _L1BTMI_S3_SUNDATA_
define _L1BTMI_S3_SUNDATA_
#endif

typedef struct {
    float solarBetaAngle;
    float phaseFromOrbitMidnight;
    float sunEarthSeparation;
    float earthAngularRadius;
    float phaseOfEclipseExit;
    float orbitRate;
    float timeSinceEclipseEntry;
```
float sunVectorInBodyFrame[3];
} L1BTMI_S3_SUNDATA;
#endif

#ifndef _L1BTMI_S3_CALCOUNTS_
#define _L1BTMI_S3_CALCOUNTS_

typedef struct {
    float hotLoadThermisterTemp[2][3];
    unsigned short hotLoadReading[2][16];
    unsigned short coldLoadReading[2][16];
} L1BTMI_S3_CALCOUNTS;
#endif

#ifndef _L1BTMI_S3_CALIBRATION_
#define _L1BTMI_S3_CALIBRATION_

typedef struct {
    float hotLoadTemp[2];
    float coldSkyTemp[2];
    float onOrbitNonLinearity[2];
    unsigned short meanHotLoadCount[2];
    unsigned short meanColdSkyCount[2];
    float gain[2][2];
    float offset[2][2];
    float nonLinearGain[2];
    short calibrationQCflag;
    float receiverTemp[2];
    float receiverGain[2];
} L1BTMI_S3_CALIBRATION;
#endif

#ifndef _L1BTMI_S3_SCANSTATUS_
#define _L1BTMI_S3_SCANSTATUS_

typedef struct {
    unsigned char dataQuality;
    unsigned char missing;
    unsigned char modeStatus;
    short geoError;
} L1BTMI_S3SCANSTATUS;
typedef struct {
    SCANTIME ScanTime;
    float Latitude[208];
    float Longitude[208];
    L1BTMI_S3_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L1BTMI_S3_CALIBRATION calibration;
    float moonVectorInstFrame[3];
    L1BTMI_S3_CALCOUNTS calCounts;
    L1BTMI_S3_SUNDATA sunData;
    float incidenceAngle[208];
    float satAzimuthAngle[208];
    float solarZenAngle[208];
    float solarAzimuthAngle[208];
    float sunGlintAngle[208];
    float Tb[208][2];
    short RFIFlag[208][1];
} L1BTMI_S3;

#define _L1BTMI_S2_SUNDATA_

typedef struct {
    // STRUCT content
} L1BTMI_S2_SUNDATA;
float solarBetaAngle;
float phaseFromOrbitMidnight;
float sunEarthSeparation;
float earthAngularRadius;
float phaseOfEclipseExit;
float orbitRate;
float timeSinceEclipseEntry;
float sunVectorInBodyFrame[3];
}

#endif

#ifndef _L1BTMI_S2_CALCOUNTS_
#define _L1BTMI_S2_CALCOUNTS_

typedef struct {
    float hotLoadThermisterTemp[5][3];
    unsigned short hotLoadReading[5][8];
    unsigned short coldLoadReading[5][8];
} L1BTMI_S2_CALCOUNTS;

#endif

#ifndef _L1BTMI_S2_CALIBRATION_
#define _L1BTMI_S2_CALIBRATION_

typedef struct {
    float hotLoadTemp[5];
    float coldSkyTemp[5];
    float onOrbitNonLinearity[5];
    unsigned short meanHotLoadCount[5];
    unsigned short meanColdSkyCount[5];
    float gain[5][2];
    float offset[5][2];
    float nonLinearGain[5];
    short calibrationQCflag;
    float receiverTemp[5];
    float receiverGain[5];
} L1BTMI_S2_CALIBRATION;

#endif

#ifndef _L1BTMI_S2_SCANSTATUS_
#define _L1BTMI_S2_SCANSTATUS_

typedef struct {
    unsigned char dataQuality;
    unsigned char missing;
    unsigned char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    unsigned char tmiIsStatus;
    double FractionalGranuleNumber;
    short attDetermSource;
    signed char TRMMcontMode;
    signed char TRMYawUpdateS;
    signed char TRMqac;
} L1BTMI_S2_SCANSTATUS;

#endif

#ifndef _L1BTMI_S2_
#define _L1BTMI_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[104];
    float Longitude[104];
    L1BTMI_S2_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L1BTMI_S2_CALIBRATION calibration;
    float moonVectorInstFrame[3];
    L1BTMI_S2_CALCOUNTS calCounts;
    L1BTMI_S2_SUNDATA sunData;
    float incidenceAngle[104];
    float satAzimuthAngle[104];
    float solarZenAngle[104];
    float solarAzimuthAngle[104];
    float sunGlintAngle[104];
    float Tb[104][5];
    short RFIFlag[104][1];
} L1BTMI_S2;
typedef struct {
   float solarBetaAngle;
   float phaseFromOrbitMidnight;
   float sunEarthSeparation;
   float earthAngularRadius;
   float phaseOfEclipseExit;
   float orbitRate;
   float timeSinceEclipseEntry;
   float sunVectorInBodyFrame[3];
} L1BTMI_S1_SUNDATA;

typedef struct {
   float hotLoadThermisterTemp[2][3];
   unsigned short hotLoadReading[2][8];
   unsigned short coldLoadReading[2][8];
} L1BTMI_S1_CALCOUNTS;

typedef struct {
   float hotLoadTemp[2];
   float coldSkyTemp[2];
   float onOrbitNonLinearity[2];
   unsigned short meanHotLoadCount[2];
   unsigned short meanColdSkyCount[2];
   float gain[2][2];
   float offset[2][2];
   float nonLinearGain[2];
   short calibrationQCflag;
} L1BTMI_S1_CALIBRATION;
float receiverTemp[2];
float receiverGain[2];
} L1BTMI_S1_CALIBRATION;
#endif

#ifndef _NAVIGATION_
define _NAVIGATION_

typedef struct {
  float scPos[3];
  float scVel[3];
  float scLat;
  float scLon;
  float scAlt;
  float dprAlt;
  float scAttRollGeoc;
  float scAttPitchGeoc;
  float scAttYawGeoc;
  float scAttRollGeod;
  float scAttPitchGeod;
  float scAttYawGeod;
  float greenHourAng;
  double timeMidScan;
  double timeMidScanOffset;
} NAVIGATION;
#endif

#ifndef _L1BTMI_S1_SCANSTATUS_
define _L1BTMI_S1_SCANSTATUS_

typedef struct {
  unsigned char dataQuality;
  unsigned char missing;
  unsigned char modeStatus;
  short geoError;
  short geoWarning;
  short SCorientation;
  short pointingStatus;
  signed char acsModeMidScan;
  signed char targetSelectionMidScan;
  unsigned char tmiIsStatus;
} L1BTMI_S1_SCANSTATUS;
#define _L1BTMI_S1_SCANSTATUS_
#endif

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

typedef struct {
    SCANTIME ScanTime;
    float Latitude[104];
    float Longitude[104];
    L1BTMI_S1_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L1BTMI_S1_CALIBRATION calibration;
    float moonVectorInstFrame[3];
    L1BTMI_S1_CALCOUNTS calCounts;
    L1BTMI_S1_SUNDATA sunData;
    float incidenceAngle[104][2];
    float satAzimuthAngle[104];
    float solarZenAngle[104];
    float solarAzimuthAngle[104];
float sunGlintAngle[104];
float Tb[104][2];
short RFIFlag[104][1];
} L1BTMI_S1;
#endif

#ifndef _L1BTMI_SWATHS_
define _L1BTMI_SWATHS_

typedef struct {
    L1BTMI_S1 S1;
    L1BTMI_S2 S2;
    L1BTMI_S3 S3;
} L1BTMI_SWATHS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L1BTMI_S3_SUNDATA/
    REAL*4 solarBetaAngle
    REAL*4 phaseFromOrbitMidnight
    REAL*4 sunEarthSeparation
    REAL*4 earthAngularRadius
    REAL*4 phaseOfEclipseExit
    REAL*4 orbitRate
    REAL*4 timeSinceEclipseEntry
    REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BTMI_S3_CALCOUNTS/
    REAL*4 hotLoadThermisterTemp(3,2)
    INTEGER*2 hotLoadReading(16,2)
    INTEGER*2 coldLoadReading(16,2)
END STRUCTURE

STRUCTURE /L1BTMI_S3_CALIBRATION/
    REAL*4 hotLoadTemp(2)
    REAL*4 coldSkyTemp(2)
    REAL*4 onOrbitNonLinearity(2)
INTEGER*2 meanHotLoadCount(2)
INTEGER*2 meanColdSkyCount(2)
REAL*4 gain(2,2)
REAL*4 offset(2,2)
REAL*4 nonLinearGain(2)
INTEGER*2 calibrationQCflag
REAL*4 receiverTemp(2)
REAL*4 receiverGain(2)
END STRUCTURE

STRUCTURE /L1BTMI_S3_SCANSTATUS/
  CHARACTER dataQuality
  CHARACTER missing
  CHARACTER modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  CHARACTER tmiIsStatus
  REAL*8 FractionalGranuleNumber
  INTEGER*2 attDetermSource
  BYTE TRMMcontMode
  BYTE TRMMyawUpdateS
  BYTE TRMMqac
END STRUCTURE

STRUCTURE /L1BTMI_S3/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(208)
  REAL*4 Longitude(208)
  RECORD /L1BTMI_S3_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L1BTMI_S3_CALIBRATION/ calibration
  REAL*4 moonVectorInstFrame(3)
  RECORD /L1BTMI_S3_CALCOUNTS/ calCounts
  RECORD /L1BTMI_S3_SUNDATA/ sunData
  REAL*4 incidenceAngle(208)
  REAL*4 satAzimuthAngle(208)
  REAL*4 solarZenAngle(208)
  REAL*4 solarAzimuthAngle(208)
  REAL*4 sunGlintAngle(208)
REAL*4 Tb(2,208)
INTEGER*2 RFIFlag(1,208)
END STRUCTURE

STRUCTURE /L1BTMI_S2_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
  REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BTMI_S2_CALCOUNTS/
  REAL*4 hotLoadThermisterTemp(3,5)
  INTEGER*2 hotLoadReading(8,5)
  INTEGER*2 coldLoadReading(8,5)
END STRUCTURE

STRUCTURE /L1BTMI_S2_CALIBRATION/
  REAL*4 hotLoadTemp(5)
  REAL*4 coldSkyTemp(5)
  REAL*4 onOrbitNonLinearity(5)
  INTEGER*2 meanHotLoadCount(5)
  INTEGER*2 meanColdSkyCount(5)
  REAL*4 gain(2,5)
  REAL*4 offset(2,5)
  REAL*4 nonLinearGain(5)
  INTEGER*2 calibrationQCflag
  REAL*4 receiverTemp(5)
  REAL*4 receiverGain(5)
END STRUCTURE

STRUCTURE /L1BTMI_S2_SCANSTATUS/
  CHARACTER dataQuality
  CHARACTER missing
  CHARACTER modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
CHARACTER tmiIsStatus
REAL*8 FractionalGranuleNumber
INTEGER*2 attDetermSource
BYTE TRMMcontMode
BYTE TRMMyawUpdateS
BYTE TRMMqac
END STRUCTURE

STRUCTURE /L1BTMI_S2/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(104)
  REAL*4 Longitude(104)
  RECORD /L1BTMI_S2_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L1BTMI_S2_CALIBRATION/ calibration
  REAL*4 moonVectorInstFrame(3)
  RECORD /L1BTMI_S2_CALCOUNTS/ calCounts
  RECORD /L1BTMI_S2_SUNDATA/ sunData
  REAL*4 incidenceAngle(104)
  REAL*4 satAzimuthAngle(104)
  REAL*4 solarZenAngle(104)
  REAL*4 solarAzimuthAngle(104)
  REAL*4 sunGlintAngle(104)
  REAL*4 Tb(5,104)
  INTEGER*2 RFIFlag(1,104)
END STRUCTURE

STRUCTURE /L1BTMI_S1_SUNDATA/
  REAL*4 solarBetaAngle
  REAL*4 phaseFromOrbitMidnight
  REAL*4 sunEarthSeparation
  REAL*4 earthAngularRadius
  REAL*4 phaseOfEclipseExit
  REAL*4 orbitRate
  REAL*4 timeSinceEclipseEntry
  REAL*4 sunVectorInBodyFrame(3)
END STRUCTURE

STRUCTURE /L1BTMI_S1_CALCOUNTS/
  REAL*4 hotLoadThermisterTemp(3,2)
  INTEGER*2 hotLoadReading(8,2)
INTEGER*2 coldLoadReading(8,2)
END STRUCTURE

STRUCTURE /L1BTMI_S1_CALIBRATION/
  REAL*4 hotLoadTemp(2)
  REAL*4 coldSkyTemp(2)
  REAL*4 onOrbitNonLinearity(2)
  INTEGER*2 meanHotLoadCount(2)
  INTEGER*2 meanColdSkyCount(2)
  REAL*4 gain(2,2)
  REAL*4 offset(2,2)
  REAL*4 nonLinearGain(2)
  INTEGER*2 calibrationQCflag
  REAL*4 receiverTemp(2)
  REAL*4 receiverGain(2)
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
  REAL*4 scAlt
  REAL*4 dprAlt
  REAL*4 scAttRollGeoc
  REAL*4 scAttPitchGeoc
  REAL*4 scAttYawGeoc
  REAL*4 scAttRollGeod
  REAL*4 scAttPitchGeod
  REAL*4 scAttYawGeod
  REAL*4 greenHourAng
  REAL*8 timeMidScan
  REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1BTMI_S1_SCANSTATUS/
  CHARACTER dataQuality
  CHARACTER missing
  CHARACTER modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
CHARACTER tmiIsStatus
REAL*8 FractionalGranuleNumber
INTEGER*2 attDetermSource
BYTE TRMMcontMode
BYTE TRMMyawUpdateS
BYTE TRMMqac
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BTMI_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(104)
  REAL*4 Longitude(104)
  RECORD /L1BTMI_S1_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L1BTMI_S1_CALIBRATION/ calibration
  REAL*4 moonVectorInstFrame(3)
  RECORD /L1BTMI_S1_CALCOUNTS/ calCounts
  RECORD /L1BTMI_S1_SUNDATA/ sunData
  REAL*4 incidenceAngle(2,104)
  REAL*4 satAzimuthAngle(104)
  REAL*4 solarZenAngle(104)
  REAL*4 solarAzimuthAngle(104)
  REAL*4 sunGlintAngle(104)
  REAL*4 Tb(2,104)
  INTEGER*2 RFIFlag(1,104)
END STRUCTURE

STRUCTURE /L1BTMI_SWATHS/
  RECORD /L1BTMI_S1/ S1;
The VIRS Level-1B Product, 1BVIRS, "VIRS Radiance," is written in HDF. The following sizing parameter is used in describing these formats:

Dimension definitions:
- nscan var Number of scans in the granule.
- npixel 261 Number of pixels in each scan.
- nchan 5 Number of channels.
- nchanvis 2 Number of visible channels.
- nchanir 3 Number of channels with infrared.

Figure 309 through Figure 313 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo (Metadata):**
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**Swath (Swath)**
Figure 309: Data Format Structure for 1BVIRS, VIRS Radiance
Figure 310: Data Format Structure for 1BVIRS, ScanTime

Figure 311: Data Format Structure for 1BVIRS, scanStatus
Figure 312: Data Format Structure for 1BVIRS, navigation

Figure 313: Data Format Structure for 1BVIRS, solarCal
SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group)

**dataQuality** (1-byte char, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError indicates bad or missing values</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not normal</td>
</tr>
<tr>
<td>7</td>
<td>QAC errors associated with this scan</td>
</tr>
</tbody>
</table>

**missing** (1-byte char, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte char, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)). The non-routine situations follow:
<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation is not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine instrument status</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0 the unsigned integer value is \( 2^i \)):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>90</td>
<td>-Y forward (yaw 90)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8002</td>
<td>Yaw turn in progress</td>
</tr>
<tr>
<td>-8003</td>
<td>Deep Space Calibration in progress</td>
</tr>
<tr>
<td>-8004</td>
<td>Non-nominal pointing other than above</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
</table>
Nominal ACS mode (4) for mission science
Non-nominal ACS mode

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
</tr>
<tr>
<td>1</td>
<td>Sun Acquire</td>
</tr>
<tr>
<td>2</td>
<td>Earth Acquire</td>
</tr>
<tr>
<td>3</td>
<td>Yaw Acquire</td>
</tr>
<tr>
<td>4</td>
<td>Nominal</td>
</tr>
<tr>
<td>5</td>
<td>Yaw Maneuver</td>
</tr>
<tr>
<td>6</td>
<td>Delta-H (Thruster)</td>
</tr>
<tr>
<td>7</td>
<td>Delta-V (Thruster)</td>
</tr>
<tr>
<td>8</td>
<td>CERES Calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Unknown -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Yaw = 0 or maneuver in progress to yaw = 0</td>
</tr>
<tr>
<td>1</td>
<td>Yaw = 180 or maneuver in progress to yaw = 180</td>
</tr>
<tr>
<td>2</td>
<td>Yaw = 90 or maneuver in progress to yaw = 90</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

virsInstS (1-byte integer, array size: nscan):

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Day (no calibration occurring)</td>
</tr>
<tr>
<td>1</td>
<td>Night</td>
</tr>
<tr>
<td>2</td>
<td>Monitor Scan Stability</td>
</tr>
<tr>
<td>3</td>
<td>Day with Calibration</td>
</tr>
</tbody>
</table>

virsMode (1-byte integer, array size: nscan):
Value Meaning
0 mission mode
1 safehold mode
2 outgas mode
3 activation mode

\textbf{virsAbnCon} (1-byte char, array size: nscan):
Bit 0 is the most significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^{8-i} - 1$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>normal</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>scan phase error</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>normal</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>selftest error</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>normal</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>thermal data missing</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>normal</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>not used</td>
</tr>
</tbody>
</table>

\textbf{FractionalGranuleNumber} (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

\textbf{attDetermSource} (2-byte integer, array size: nscan):

Attitude determination source.
A flag explaining how the attitude value was calculated.
Improved estimates make use of ground processing of PR science-instrument-measured roll values, Gyroscope data, and Sun Sensor 1 data. Earlier products (TRMM V7 and before) used the onboard attitudes with various corrections.
Values were determined for each granule based on the data available and conditions for each orbit. Flag values follow.
Value  Meaning
430 and higher Best accuracy, good data for this orbit
421 Reduced accuracy, PR roll data not available (affecting roll/yaw estimates)
413 Reduced accuracy, sun data not available (affecting pitch)
411 Reduced accuracy, PR roll and sun sensor not available
300-399 Reduced accuracy due to various special conditions
200-299 Fallback to using the onboard attitude estimates with TRMM V7 corrections
-91 Spacecraft in safehold mode, no science data
-99 No data due to telemetry data gap

TRMM_contMode (1-byte integer, array size: nscan):

The Contingency Mode Flag from telemetry indicates alternate attitude control of the spacecraft. The nominal at-launch Attitude Control System (ACS) for TRMM used Earth horizon sensors for pitch and roll control, and the yaw was updated twice each orbit using the Sun Sensors and propagated using gyro data. However, due to possible problems identified with the Earth Sensor Assembly (ESA) lifetime on-orbit, a contingency ACS mode was developed late in the development cycle. This mode used the Sun Sensors, magnetometers, and gyroscope data. It proved very valuable when the horizon sensors had problems with TRMM moving to the higher operating altitude (from 350 to 402.5 km) to extend the mission lifetime. Thus the contingency mode was used throughout the post-boost period. It was also tested early in the mission on 1998-01-13.

Value  Meaning
0 Nominal control of spacecraft used in the pre-boost period
1 Contingency mode control used in the post-boost period
-99 Missing

TRMM_yawUpdateS (1-byte integer, array size: nscan):

The Yaw Update Status flag in telemetry gives the status of the Yaw accuracy for the nominal pre-boost Attitude Control
System (ACS) operation. The yaw is considered "indeterminate" in various non-nominal control modes, and after the return to the nominal Earth pointing (using the Earth sensor for pitch and roll), the yaw is considered "inaccurate" until the time when an "update" is done using a Sun sensor (at certain positions in the orbit). Before the update "the yaw attitude knowledge is acceptable for ACS use, but might not be acceptable for science use" according to ACS Software User’s Guide.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Inaccurate</td>
</tr>
<tr>
<td>1</td>
<td>Indeterminate</td>
</tr>
<tr>
<td>2</td>
<td>Accurate</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

TRMMqac (1-byte integer, array size: nscan):
The Quality and Accounting Capsule of the Science packet as it appears in Level-0 data. If no QAC is given in Level-0, which means no decoding errors occurred, QAC in this format has a value of zero.

navigation (Group)

scPos (4-byte float, array size: XYZ x nscan):
The position vector(m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector (ms\(^{-1}\)) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values
range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**solarCal** (Group)

**sunVecX** (8-byte float, array size: nscan):
Solar Position (X-component) (Geocentric Inertial Coord).

**sunVecY** (8-byte float, array size: nscan):
Solar Position (Y-component) (Geocentric Inertial Coord).

**sunVecZ** (8-byte float, array size: nscan):
Solar Position (Z-component) (Geocentric Inertial Coord).

**sunMag** (8-byte float, array size: nscan):
Sun-Earth Distance (m).

**incidenceAngle** (4-byte float, array size: npixel x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**satAzimuthAngle** (4-byte float, array size: npixel x nscan):
The angle clockwise looking down between the local pixel geodetic north and the direction
to the satellite. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarZenAngle** (4-byte float, array size: n_pixel x n_scan):
The angle between the local pixel geodetic zenith and the direction to the sun. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**solarAzimuthAngle** (4-byte float, array size: n_pixel x n_scan):
The angle clockwise looking down between the local pixel geodetic north and the direction to the sun. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (4-byte float, array size: n_pixel x n_scan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. More specifically, define a Sun Vector from the viewed pixel location on the earth ellipsoid-model surface to the sun. Also define an Inverse Satellite Vector from the pixel to the satellite. Then reflect the Inverse Satellite Vector off the earth’s surface at the pixel location to form the Reflected Satellite View Vector. **sunGlintAngle** is the angular separation between the Reflected Satellite View Vector and the Sun Vector. When **sunGlintAngle** is zero, the instrument views the center of the specular (mirror-like) sun reflection. Values range from 0 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**calCounts** (2-byte integer, array size: 5 x 2 x 3 x n_scan):
Raw calibration counts are given in four dimensions. The first dimension is the channel number, the second dimension is the data word, the third dimension is blackbody, space view and solar diffuser, in that order, and the fourth dimension is the number of scans.

**tempCounts** (2-byte integer, array size: 6 x n_scan):
Temperatures of the black body, primary and redundant, the radiant cooler temperatures, primary and redundant, the mirror temperature, and the electronics module temperature. All quantities have units of counts, and have minimum values of 0, and maximum values of 4095.

**radiance** (4-byte float, array size: n_chan x n_pixel x n_scan):
Scene data for the channels, measured in Radiance \( (mW cm^{-2} \mu m^{-1} sr^{-1}) \). sr means steradian. The three dimensions are channel, pixel, and scan. The range, accuracy and wavelength for each channel are as follows:

<table>
<thead>
<tr>
<th>Channel</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Accuracy</th>
<th>Wavelength (micrometers)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>65.5</td>
<td>10%</td>
<td>0.63</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>32.7</td>
<td>10%</td>
<td>1.6</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0.111</td>
<td>2%</td>
<td>3.75</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1.371</td>
<td>2%</td>
<td>10.8</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1.15</td>
<td>2%</td>
<td>12.0</td>
</tr>
</tbody>
</table>
reflectance (4-byte float, array size: nchanvis x npixel x nscan):  
Scene data for channels 1 and 2, measured in reflectance. The three dimensions are channel, pixel, and scan.

Tb (4-byte float, array size: nchanir x npixel x nscan):  
Scene data for channels 3, 4 and 5, measured in brightness temperature (K). The three dimensions are channel, pixel, and scan.

C Structure Header file:

```c
#ifndef _TK_1BVIRS_H_
define _TK_1BVIRS_H_
#endif

#ifndef _L1BVIRS_SOLARCAL_
define _L1BVIRS_SOLARCAL_
#endif

typedef struct {
    double sunVecX;
    double sunVecY;
    double sunVecZ;
    double sunMag;
} L1BVIRS_SOLARCAL;
#endif

#ifndef _NAVIGATION_
define _NAVIGATION_
#endif

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;
```
typedef struct {
    unsigned char dataQuality;
    unsigned char missing;
    unsigned char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char virsInstS;
    signed char virsMode;
    unsigned char virsAbnCon;
    double FractionalGranuleNumber;
    short attDetermSource;
    signed char TRMMcontMode;
    signed char TRMMyawUpdateS;
    signed char TRMMqac;
} L1BVIRS_SCANSTATUS;

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
typedef struct {
    SCANTIME ScanTime;
    float Latitude[261];
    float Longitude[261];
    L1BVIRS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L1BVIRS_SOLARCAL solarCal;
    float incidenceAngle[261];
    float satAzimuthAngle[261];
    float solarZenAngle[261];
    float solarAzimuthAngle[261];
    float sunGlintAngle[261];
    short calCounts[3][2][5];
    short tempCounts[6];
    float radiance[261][5];
    float reflectance[261][2];
    float Tb[261][3];
} L1BVIRS_SWATH;

Fortran Structure Header file:

STRUCTURE /L1BVIRS_SOLARCAL/
    REAL*8 sunVecX
    REAL*8 sunVecY
    REAL*8 sunVecZ
    REAL*8 sunMag
END STRUCTURE

STRUCTURE /NAVIGATION/
    REAL*4 scPos(3)
    REAL*4 scVel(3)
    REAL*4 scLat
    REAL*4 scLon
REAL*4 scAlt
REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1BVIRS_SCANSTATUS/
  CHARACTER dataQuality
  CHARACTER missing
  CHARACTER modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE virsInstS
  BYTE virsMode
  CHARACTER virsAbnCon
  REAL*8 FractionalGranuleNumber
  INTEGER*2 attDetermSource
  BYTE TRMMyawMode
  BYTE TRMMyawUpdateS
  BYTE TRMMqc
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
5.21 1CTMI - GPM Common Calibrated Brightness Temperature

1CTMI contains common calibrated brightness temperatures from the TMI passive microwave instrument flown on the TRMM satellite. There are 3 swaths. Swath S1 has 2 low resolution channels (10V 10H). Swath S2 has 5 low resolution channels (19V 19H 21V 37V 37H). Swath S3 has 2 high resolution channels (85V 85H). Data for all swaths is observed in the same revolution of the instrument.

Earth observations are taken during a segment of the rotation when TMI is looking in the +x direction of the TRMM satellite. Since the spacecraft turns around every few weeks, +x may be forward or aft. We define the spacecraft axis v, used in the definition of the variable SCorientation, at the center of this segment and the same as the +x direction.

RELATION BETWEEN THE SWATHS: Swath S2 has the same number of scans and pixels as Swath S1. Swath S3 has the same number of scans but twice as many pixels as Swath S1. Each S1 and S2 scan contains low frequency channels sampled 104 times along the scan. Each S3 scan contains high frequency channels sampled 208 times along the scan. S1 S2 and S3 scans are repeated every 1.9s. Along an S1 scan every other center of an S3 pixel coincides with the center of an S1 pixel.

The Figure below shows the locations of the pixels of scans 1 and 2 for Swath 1 and Swath 3. Each “+” represents centers of pixels from one or more swaths. For example, the label
"S1:1,2 S3:1,3" means that both Swath S1, Scan 1, Pixel 2 and Swath S3, Scan 1, Pixel 3 are located at the "+".

\[
\begin{align*}
S1:1,1 & \quad S3:1,1 & \quad S3:1,2 & \quad S1:1,2 & \quad S3:1,3 & \quad S1:1,104 & \quad S3:1,207 & \quad S3:1,208 \\
+ & \quad + & \quad + & \quad \ldots & \quad + & \quad + \\
S1:2,1 & \quad S3:2,1 & \quad S3:2,2 & \quad S1:2,2 & \quad S3:2,3 & \quad S1:2,104 & \quad S3:2,207 & \quad S3:2,208 \\
+ & \quad + & \quad + & \quad \ldots & \quad + & \quad +
\end{align*}
\]

Dimension definitions:
- nscan1 var Number of Swath S1 scans in the granule.
- nchannel1 2 Number of Swath S1 channels (10V 10H).
- npixel1 104 Number of Swath S1 pixels in one scan.
- nchUIA1 2 Number of Swath S1 unique incidence angles.
- nscan2 var Number of Swath S3 scans in the granule.
- nchannel2 5 Number of Swath S2 channels (19V 19H 21V 37V 37H).
- npixel2 104 Number of Swath S2 pixels in one scan.
- nchUIA2 1 Number of Swath S2 unique incidence angles.
- nscan3 var Number of Swath S3 scans in the granule.
- nchannel3 2 Number of Swath S3 channels (85V 85H).
- npixel3 208 Number of Swath S3 pixels in one scan.
- nchUIA3 1 Number of Swath S3 unique incidence angles.

Figure 314 through Figure 323 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
Figure 314: Data Format Structure for 1CTMI, GPM Common Calibrated Brightness Temperature

Figure 315: Data Format Structure for 1CTMI, S1
Figure 316: Data Format Structure for 1CTMI, S2

Figure 317: Data Format Structure for 1CTMI, S3
5 STANDARD GPM PRODUCTS

Figure 318: Data Format Structure for 1CTMI, S1, ScanTime

Figure 319: Data Format Structure for 1CTMI, S1, SCstatus

Figure 320: Data Format Structure for 1CTMI, S2, ScanTime
5.21 1CTMI - GPM Common Calibrated Brightness Temperature

Figure 321: Data Format Structure for 1CTMI, S2, SCstatus

Figure 322: Data Format Structure for 1CTMI, S3, ScanTime

Figure 323: Data Format Structure for 1CTMI, S3, SCstatus
FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

XCALinfo (Metadata):
XCALinfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

S1 (Swath)

S1_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S1_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S1)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value
MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group in S1)

SCorientation (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value
**FractionalGranuleNumber** (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**Quality** (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
    1-99 = Generic flags (all sensors)
    100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
    -(1-98) = Generic flags (all sensors)
    -99 = Missing value (no quality information available)
    -(100-127) = Sensor specific flags

**DETAILED SPECIFICATIONS:**
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion
-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km
    threshold. used in L1C-R product only
-99 = Missing value (no quality information available)
-100 = Quality and Accounting Capsule errors this scan

**incidenceAngle** (4-byte float, array size: nchUL x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel
location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9  Missing value
sunGlintAngle (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel1 x nscan1):
Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

\[
\text{incidenceAngle}(2, \text{npixel}, \text{nscan}) \\
\text{sunGlintAngle}(2, \text{npixel}, \text{nscan}) \\
\text{incidenceAngleIndex}(10, \text{nscan}) \\
\text{Tc}(10, \text{npixel}, \text{nscan})
\]

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
i = \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
i_a = \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
sga = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99   Missing value

Tc (4-byte float, array size: nchannel1 x npixel1 x nscan1):

GPM Common Calibrated Brightness Temperature. The channels are:
10.65 GHz vertically-polarized TBs
10.65 GHz horizontally-polarized TBs
Values range from 0 to 10000 K. Special values are defined as:
-9999.9  Missing value

S2 (Swath)

S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S2_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

Month (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value
DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: npixel2 x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: npixel2 x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCstatus (Group in S2)

SCorientation (2-byte integer, array size: nscan2):
The angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. The relationship of v to the sensor geometry is defined
in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values
are defined as:
-9999  Missing value

SClatitude (4-byte float, array size: nscan2):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

SClongitude (4-byte float, array size: nscan2):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCaltitude (4-byte float, array size: nscan2):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9  Missing value

FractionalGranuleNumber (8-byte float, array size: nscan2):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**Quality** (1-byte integer, array size: npixel2 x nscan2):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**
- 0 = Good data in all channels in the swath
- gt 0 = Cautionary warning flags
  - 1-99 = Generic flags (all sensors)
  - 100-127 = Sensor specific flags
- lt 0 = Major errors resulting in missing data
  - -(1-98) = Generic flags (all sensors)
  - -99 = Missing value (no quality information available)
  - -(100-127) = Sensor specific flags

**DETAILED SPECIFICATIONS:**
- 1 = Possible sunGlint, 0 ≤ sunGlintAngle ≤ 20
- 2 = Possible radio frequency interference
- 3 = Degraded geolocation data
- 4 = Data corrected for warm load intrusion
- -1 = Data is missing from file or unreadable, missing scan
- -2 = Invalid Tb or unphysical brightness temperature Tb ≤ 50 or Tb ≥ 350
- -3 = Error in geolocation
- -4 = Data is missing in 1 channel
- -5 = Data is missing in multiple channels
- -6 = Lat/Lon values are out of range
- -7 = Non-normal status modes
- -10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
- -99 = Missing value (no quality information available)
- -100 = Quality and Accounting Capsule errors this scan

**incidenceAngle** (4-byte float, array size: nchUIA2 x npixel2 x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA2 x npixel2 x nscan2):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.
**incidenceAngleIndex** (1-byte integer, array size: nchannel2 x nsan2):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

```
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nsan)
Tc(10,npixel,nscan)
```

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```
i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)
```

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:

-99  Missing value

**Tc** (4-byte float, array size: nchannel2 x npixel2 x nsan2):

GPM Common Calibrated Brightness Temperature. The channels are:

- 19.35 GHz vertically-polarized TBs
- 19.35 GHz horizontally-polarized TBs
- 21.3 GHz vertically-polarized TBs
- 37.0 GHz vertically-polarized TBs
- 37.0 GHz horizontally-polarized TBs

Values range from 0 to 10000 K. Special values are defined as:

-9999.9  Missing value
S3 (Swath)

**S3_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S3_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S3)
A UTC time associated with the scan.

- **Year** (2-byte integer, array size: nscan3):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
  - 9999 Missing value

- **Month** (1-byte integer, array size: nscan3):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
  - 99 Missing value

- **DayOfMonth** (1-byte integer, array size: nscan3):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
  - 99 Missing value

- **Hour** (1-byte integer, array size: nscan3):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
  - 99 Missing value

- **Minute** (1-byte integer, array size: nscan3):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
  - 99 Missing value

- **Second** (1-byte integer, array size: nscan3):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
  - 99 Missing value

- **MilliSecond** (2-byte integer, array size: nscan3):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
  - 9999 Missing value

- **DayOfYear** (2-byte integer, array size: nscan3):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
  - 9999 Missing value
SecondOfDay (8-byte float, array size: nscan3):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel3 x nscan3):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel3 x nscan3):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group in S3)

SCorientation (2-byte integer, array size: nscan3):
The angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. The relationship of v to the sensor geometry is defined
in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values
are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan3):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan3):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan3):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan3):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel3 x nscan3):
Quality of Tc in the swath.
GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-1-98 = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion
-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)
-100 = Quality and Accounting Capsule errors this scan

**incidenceAngle** (4-byte float, array size: nchUIA3 x npixel3 x nscan3):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA3 x npixel3 x nscan3):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel3 x nscan3):
Index (1 based as in Fortran) of the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
i = \text{incidenceAngleIndex}(\text{channel}, \text{scan})
\]
\[
\text{ia} = \text{incidenceAngle}(i, \text{pixel}, \text{scan})
\]
\[
\text{sga} = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

\text{Tc} (4-byte float, array size: nchannel3 x npixel3 x nscan3):

GPM Common Calibrated Brightness Temperature. The channels are:
85.5 GHz vertically-polarized TBs
85.5 GHz horizontally-polarized TBs

Values range from 0 to 400 K. Special values are defined as:
-9999.9 Missing value

\text{C Structure Header file:}

```c
#ifndef _TK_1CTMI_H_
#define _TK_1CTMI_H_

#ifndef _L1CTMI_S3_
```
#define _L1CTMI_S3_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[208];
    float Longitude[208];
    SCSTATUS SCstatus;
    signed char Quality[208];
    float incidenceAngle[208][1];
    signed char sunGlintAngle[208][1];
    signed char incidenceAngleIndex[2];
    float Tc[208][2];
} L1CTMI_S3;
#endif

#ifndef _L1CTMI_S2_
#define _L1CTMI_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[104];
    float Longitude[104];
    SCSTATUS SCstatus;
    signed char Quality[104];
    float incidenceAngle[104][1];
    signed char sunGlintAngle[104][1];
    signed char incidenceAngleIndex[5];
    float Tc[104][5];
} L1CTMI_S2;
#endif

#ifndef _SCSTATUS_
#define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
#endif
typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

typedef struct {
    SCANTIME ScanTime;
    float Latitude[104];
    float Longitude[104];
    SCSTATUS SCstatus;
    signed char Quality[104];
    float incidenceAngle[104][2];
    signed char sunGlintAngle[104][2];
    signed char incidenceAngleIndex[2];
    float Tc[104][2];
} L1CTMI_S1;

typedef struct {
    L1CTMI_S1 S1;
    L1CTMI_S2 S2;
L1CTMI_S3 S3;
} L1CTMI_SWATHS;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L1CTMI_S3/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(208)
  REAL*4 Longitude(208)
  RECORD /SCSTATUS/ SCstatus
  BYTE Quality(208)
  REAL*4 incidenceAngle(1,208)
  BYTE sunGlintAngle(1,208)
  BYTE incidenceAngleIndex(2)
  REAL*4 Tc(2,208)
END STRUCTURE

STRUCTURE /L1CTMI_S2/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(104)
  REAL*4 Longitude(104)
  RECORD /SCSTATUS/ SCstatus
  BYTE Quality(104)
  REAL*4 incidenceAngle(1,104)
  BYTE sunGlintAngle(1,104)
  BYTE incidenceAngleIndex(5)
  REAL*4 Tc(5,104)
END STRUCTURE

STRUCTURE /SCSTATUS/
  INTEGER*2 SCorientation
  REAL*4 SClatitude
  REAL*4 SClongitude
  REAL*4 SCaltitude
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
5.22 1CGMI - GPM Common Calibrated Brightness Temperature

1CGMI contains common calibrated brightness temperatures from the GMI passive microwave instrument flown on the GPM satellite. 1C-R GMI is a remapped version of 1CGMI which is explained at the end of this section. Both 1CGMI and 1C-R GMI have the same format. Swath S1 has 9 channels which are similar to TRMM TMI (10V 10H 19V 19H 23V 37V 37H 89V 89H). Swath S2 has 4 channels similar to AMSU-B (166V 166H 183+/-3V 183+/-7V). Data for both swaths is observed in the same revolution of the instrument.

Earth observations are taken during a segment of the rotation when GMI is looking in the +x direction of the GPM satellite. Since the spacecraft turns around every few weeks, +x may be forward or aft. We define the spacecraft axis v, used in the definition of the variable SCorientation, at the center of this segment and the same as the +x direction. 32rpm * 1min/60s * 5538s/orbit = 2954 scans / orbit.
RELATION BETWEEN THE SWATHS: Swath S2 has the same number of scans and the same number of pixels as Swath S1. Each S1 scan contains 9 channels sampled 221 times along the scan. Each S2 scan contains 4 channels sampled 221 times along the scan. Since the incidence angle of Swath S1 is different than Swath S2, the geolocations of the pixel centers are different.

1C-R GMI is a remapped version of 1CGMI. 1C-R is the input for Gprof. The 1C-R Swath S1 is the same as the 1C Swath S1. However, the 1C-R Swath S2 consists of pixels selected from 1C Swath S2 to be as close as possible to the S1 pixels. The 1C-R S2 pixels will often be observed at a different scantime and sometimes from a different granule than the corresponding S1 pixel. Since 1C S2 is narrower than 1C S1, 1C-R S2 has missing pixels on both edges of the swath.

Dimension definitions:

- $n_{scan1}$: Typical number of Swath S1 scans in the granule.
- $n_{channel1}$: Number of Swath S1 channels (10V 10H 19V 19H 23V 37V 37H 89V 89H).
- $n_{pixel1}$: Number of Swath S1 pixels in one scan.
- $n_{chUIA1}$: Number of Swath S1 unique incidence angles.
- $n_{scan2}$: Typical number of Swath S2 scans in the granule.
- $n_{channel2}$: Number of Swath S2 channels (165V 165H 183+/−3V 183+/−7V).
- $n_{pixel2}$: Number of Swath S2 pixels in one scan.
- $n_{chUIA2}$: Number of Swath S2 unique incidence angles.

Figure 324 through Figure 330 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.
5.22 1CGMI - GPM Common Calibrated Brightness Temperature

**Figure 325: Data Format Structure for 1CGMI, S1**

- **S1_SwathHeader** Metadata
- **S1_IncidenceAngleIndex** Metadata
- **ScanTime** 19 bytes Group: nschan
- **Latitude** 4 bytes Array: npixel1 x nschan
- **Longitude** 4 bytes Array: npixel1 x nschan
- **SCstatus** 22 bytes Group: nschan
- **Quality** 1 byte Array: npixel1 x nschan
- **incidenceAngle** 4 bytes Array: nchan x npixel1 x nschan
- **sunGlintAngle** 1 byte Array: nchan x npixel1 x nschan
- **incidenceAngleIndex** 1 byte Array: nchan x nschan
- **Tc** 4 bytes Array: nchan x npixel1 x nschan

**Figure 326: Data Format Structure for 1CGMI, S2**

- **S2_SwathHeader** Metadata
- **S2_IncidenceAngleIndex** Metadata
- **ScanTime** 19 bytes Group: nschan
- **Latitude** 4 bytes Array: npixel2 x nschan
- **Longitude** 4 bytes Array: npixel2 x nschan
- **SCstatus** 22 bytes Group: nschan
- **Quality** 1 byte Array: npixel2 x nschan
- **incidenceAngle** 4 bytes Array: nchan x npixel2 x nschan
- **sunGlintAngle** 1 byte Array: nchan x npixel2 x nschan
- **incidenceAngleIndex** 1 byte Array: nchan x nschan
- **Tc** 4 bytes Array: nchan x npixel2 x nschan
Figure 327: Data Format Structure for 1CGMI, S1, ScanTime

Figure 328: Data Format Structure for 1CGMI, S1, SCstatus

Figure 329: Data Format Structure for 1CGMI, S2, ScanTime
InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

XCALinfo (Metadata):
XCALinfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

S1 (Swath)

S1_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S1_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S1)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined.
as:
    -9999  Missing value

**Month** (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
    -99  Missing value

**DayOfMonth** (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
    -99  Missing value

**Hour** (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
    -99  Missing value

**Minute** (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
    -99  Missing value

**Second** (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
    -99  Missing value

**MilliSecond** (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
    -9999  Missing value

**DayOfYear** (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
    -9999  Missing value

**SecondOfDay** (8-byte float, array size: nscan1):
A time associated with the scan. scanTime.sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
    -9999.9  Missing value

**Latitude** (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
    -9999.9  Missing value

**Longitude** (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
    -9999.9  Missing value
SCstatus (Group in S1)

SCOrientation (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999  Missing value

SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9  Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

Quality (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-(1-98) = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion
100 = Scan blanking on

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

**incidenceAngle** (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:

-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

```
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
```

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```
i = incidenceAngleIndex(channel,scan)
i = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)
```
The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99: Missing value

Tc (4-byte float, array size: nchannel1 x npixel1 x nscan1):

GPM Common Calibrated Brightness Temperature.
The channels are:

10.65 GHz vertically-polarized TBs
10.65 GHz horizontally-polarized TBs
18.7 GHz vertically-polarized TBs
18.7 GHz horizontally-polarized TBs
23.8 GHz vertically-polarized TBs
36.64 GHz vertically-polarized TBs
36.64 GHz horizontally-polarized TBs
89.0 GHz vertically-polarized TBs
89.0 GHz horizontally-polarized TBs

Values range from 0 to 10000 K. Special values are defined as:
-9999.9: Missing value

S2 (Swath)

S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S2_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.
ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel2 x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel2 x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCstatus** (Group in S2)

**SCorientation** (2-byte integer, array size: nscan2):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

**SClatitude** (4-byte float, array size: nscan2):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**SClongitude** (4-byte float, array size: nscan2):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCaltitude** (4-byte float, array size: nscan2):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan2):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**Quality** (1-byte integer, array size: npixel2 x nscan2):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
  1-99 = Generic flags (all sensors)
  100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
  -(1-98) = Generic flags (all sensors)
  -99 = Missing value (no quality information available)
  -(100-127) = Sensor specific flags
DETAILED SPECIFICATIONS:

1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion
100 = Scan blanking on

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in LiC-R product only
-99 = Missing value (no quality information available)

\textbf{incidenceAngle} (4-byte float, array size: nchUIA2 x npixel2 x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:

-9999.9 Missing value

\textbf{sunGlintAngle} (1-byte integer, array size: nchUIA2 x npixel2 x nscan2):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

\textbf{incidenceAngleIndex} (1-byte integer, array size: nchannel2 x nscan2):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

\begin{verbatim}
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
\end{verbatim}

The user would do the following to retrieve the angles for a given channel, pixel, and scan:
i = incidenceAngleIndex(channel, scan)
ia = incidenceAngle(i, pixel, scan)
sga = sunGlintAngle(i, pixel, scan)

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99  Missing value

Tc (4-byte float, array size: nchannel2 x npixel2 x nscan2):

GPM Common Calibrated Brightness Temperature.
The channels are:

166.0 GHz vertically-polarized  TBs
166.0 GHz horizontally-polarized  TBs
183.31+/-3 GHz vertically-polarized  TBs
183.31+/-7 GHz vertically-polarized  TBs

Values range from 0 to 400 K. Special values are defined as:
-9999.9  Missing value

C Structure Header file:

```c
#ifndef _TK_1CGMI_H_
#define _TK_1CGMI_H_

#ifndef _L1CGMI_S2_
#define _L1CGMI_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[221];
    float Longitude[221];
    SCSTATUS SCstatus;
} L1CGMI_S2;

#endif
#endif
```
signed char Quality[221];
float incidenceAngle[221][1];
signed char sunGlintAngle[221][1];
signed char incidenceAngleIndex[4];
float Tc[221][4];
} L1CGMI_S2;

#endif

#ifndef _SCSTATUS_
define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L1CGMI_S1_
define _L1CGMI_S1_

typedef struct {
SCANTIME ScanTime;
float Latitude[221];
float Longitude[221];
SCSTATUS SCstatus;
signed char Quality[221];
float incidenceAngle[221][1];
signed char sunGlintAngle[221][1];
signed char incidenceAngleIndex[9];
float Tc[221][9];
} L1CGMI_S1;

#endif

#ifndef _L1CGMI_SWATHS_
#define _L1CGMI_SWATHS_

typedef struct {
    L1CGMI_S1 S1;
    L1CGMI_S2 S2;
} L1CGMI_SWATHS;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L1CGMI_S2/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(221)
    REAL*4 Longitude(221)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(221)
    REAL*4 incidenceAngle(1,221)
    BYTE sunGlintAngle(1,221)
    BYTE incidenceAngleIndex(4)
    REAL*4 Tc(4,221)
END STRUCTURE

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude

REAL*4 SCaltitude
REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
INTEGER*2 Year
BYTE Month
BYTE DayOfMonth
BYTE Hour
BYTE Minute
BYTE Second
INTEGER*2 MilliSecond
INTEGER*2 DayOfYear
REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1CGMI_S1/
RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(221)
REAL*4 Longitude(221)
RECORD /SCSTATUS/ SCstatus
BYTE Quality(221)
REAL*4 incidenceAngle(1,221)
BYTE sunGlintAngle(1,221)
BYTE incidenceAngleIndex(9)
REAL*4 Tc(9,221)
END STRUCTURE

STRUCTURE /L1CGMI_SWATHS/
RECORD /L1CGMI_S1/ S1;
RECORD /L1CGMI_S2/ S2;
END STRUCTURE

5.23 1CSSMI - Common Calibrated Brightness Temperature

1CSSMI contains common calibrated brightness temperature from the SSM/I passive microwave instruments flown on the DMSP satellites. Swath S1 has 5 low frequency channels (19V 19H 22V 37V 37H). Swath S2 has 2 high frequency channels (85V 85H). Earth observations for both swaths are taken during a 102.4° segment of the instrument rotation when SSM/I is looking in the aft direction from satellite F8 or the forward direction from satellites F10 - F15. We define the spacecraft vector (v) at the center of this segment. "v" is used in the definition of the variable SCorientation.
RELATION BETWEEN THE SWATHS: Each S1 scan contains low frequency channels sampled 64 times along the scan. Each S2 scan contains high frequency channels sampled 128 times along the scan. Swath S2 has exactly twice as many scans as Swath S1. S1 scans 1, 2, 3, ... coincide with S2 scans 1, 3, 5, ... S1 scans are repeated every 3.8s; S2 scans are repeated every 1.9s. Along an S1 scan every other center of an S2 sample coincides with the center of an S1 sample.

The Figure below shows the locations of the samples of Swath S1 scan 1 and Swath S2 scans 1 and 2. Each "+" represents centers of samples from one or both swaths. For example, the label "S1:1,2 S2:1,3" means that both Swath S1, scan 1, sample 2, and Swath S2, scan 1, sample 3 are located at the "+".

<table>
<thead>
<tr>
<th>S1:1,1</th>
<th>S2:1,1</th>
<th>S2:1,2</th>
<th>S1:1,2</th>
<th>S2:1,3</th>
<th>S1:1,64</th>
<th>S2:1,127</th>
<th>S2:1,128</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>+</td>
<td>+</td>
<td>........</td>
<td>+</td>
<td>+</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S2:2,1</th>
<th>S2:2,2</th>
<th>S2:2,3</th>
<th>S2:2,127</th>
<th>S2:2,128</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>+</td>
<td>+</td>
<td>........</td>
<td>+</td>
</tr>
</tbody>
</table>

KNOWN PROBLEMS OR ISSUES WITH DATA:

1. F15 data: On August 14, 2006 two radar calibration (i.e. RADCAL) beacons operating at 150 and 400 MHz were activated on board the DMSP F15 spacecraft. These beacons were found to interfere with the 22.235 GHz vertically polarized channel and the 85.5 GHz horizontally polarized channel on the SSM/I sensor. The interference to these two channels was found to vary across the scan, but it appears to very stable and correctable. A correction to the Level 1C brightness temperatures was implemented based on a comparison of mean brightness temperatures over a four month period (Sep-Dec, 2006) with data from Sep-Dec 2005. The correction to the 22V channel varies between 5 and 13 K while the correction to the 85.5H varies from 0.5 to 1.0 K. Rainfall estimates from the corrected brightness temperatures appear very consistent with those from F13 and F14, however, caution is warranted in using the RADCAL corrected data. For additional updated information on this issue please refer to the Level 1C web site (http://mrain.atmos.colostate.edu/LEVEL1C).

Dimension definitions:

- **nscan1**: Number of Swath S1 scans in the granule.
- **nchannel1**: Number of Swath S1 channels (19V 19H 22V 37V 37H).
- **npixel1**: Number of Swath S1 pixels in one scan.
- **nchUIA1**: Number of Swath S1 unique incidence angles.
- **nscan2**: Number of Swath S2 scans in the granule.
- **nchannel2**: Number of Swath S2 channels (85V 85H).
- **npixel2**: Number of Swath S2 pixels in one scan.
- **nchUIA2**: Number of Swath S2 unique incidence angles.
Figure 331: Data Format Structure for 1CSSMI, Common Calibrated Brightness Temperature

Figure 331 through Figure 337 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**XCALinfo** (Metadata):
XCALinfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

**S1** (Swath)

**S1_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
5.23 1CSSMI - Common Calibrated Brightness Temperature

Figure 332: Data Format Structure for 1CSSMI, S1

Figure 333: Data Format Structure for 1CSSMI, S2
Figure 334: Data Format Structure for 1CSSMI, S1, ScanTime

Figure 335: Data Format Structure for 1CSSMI, S1, SCstatus

Figure 336: Data Format Structure for 1CSSMI, S2, ScanTime
5.23 1CSSMI - Common Calibrated Brightness Temperature

S1.IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S1)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value
DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCstatus (Group in S1)

SOrIENTATION (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. The relationship of v to the sensor geometry is defined
in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values
are defined as:
-9999  Missing value

SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9  Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**Quality** (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**
- **0** = Good data in all channels in the swath
- **gt 0** = Cautionary warning flags
  - 1-99 = Generic flags (all sensors)
  - 100-127 = Sensor specific flags
- **lt 0** = Major errors resulting in missing data
  - -(1-98) = Generic flags (all sensors)
  - -99 = Missing value (no quality information available)
  - -(100-127) = Sensor specific flags

**DETAILED SPECIFICATIONS:**
- **1** = Possible sunGlint, 0 le sunGlintAngle lt 20
- **2** = Possible radio frequency interference
- **3** = Degraded geolocation data
- **4** = Data corrected for warm load intrusion
- **102** = Climatology check warning (19V Channel)
- **103** = Climatology check warning (19H Channel)
- **104** = Climatology check warning (22V Channel)
- **105** = Climatology check warning (37V Channel)
- **106** = Climatology check warning (37H Channel)
- **107** = Climatology check warning (19V Channel)
- **108** = Climatology check warning (19V Channel)
- **109** = Climatology check warning (Multiple low-res channels)
- **110** = Climatology check warning (Multiple high-res channels)
- **111** = Warning adjacent/cross-pol pixel flagged as bad
- **112** = Warning of increased noise in 85V channel on DMSP F08
- **113** = RADCAL correction applied to Tb22v (do not use for climate)
- **114** = Ta correction made by eliminating spikes in scan cal data
- **-1** = Data is missing from file or unreadable, missing scan
- **-2** = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
- **-3** = Error in geolocation
- **-4** = Data is missing in 1 channel
- **-5** = Data is missing in multiple channels
- **-6** = Lat/Lon values are out of range
- **-7** = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

-102 = Climatology check flagged in input BASE file
-103 = Climatology check failed (19V Channel)
-104 = Climatology check failed (19H Channel)
-105 = Climatology check failed (22V Channel)
-106 = Climatology check failed (37V Channel)
-107 = Climatology check failed (37H Channel)
-108 = Climatology check failed (85V Channel)
-109 = Climatology check failed (85H Channel)
-110 = Climatology check failed (Multiple low-res channels)
-111 = Climatology check failed (Multiple high-res channels)
-112 = Distance between pixels is nonphysical
-115 = Failure of 85V channel on DMSP F08
-116 = Failure of 85V and increased noise in 85H on DMSP F08
-117 = Failure of both 85V and 85H channels on DMSP F08
-118 = Invalid scan time
-119 = Ta set to missing due to bad cal data
-120 = All data set to missing

\textbf{incidenceAngle} (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:

-9999.9 Missing value

\textbf{sunGlintAngle} (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

\textbf{incidenceAngleIndex} (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

\begin{verbatim}
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
\end{verbatim}
The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
\begin{align*}
i & = \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
ia & = \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
s\text{ga} & = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\end{align*}
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

Tc (4-byte float, array size: nchannel1 x npixel1 x nscan1):
GPM Common Calibrated Brightness Temperature. The channels are:

- 19.35 GHz vertically-polarized TBs
- 19.35 GHz horizontally-polarized TBs
- 22.235 GHz vertically-polarized TBs
- 37.0 GHz vertically-polarized TBs
- 37.0 GHz horizontally-polarized TBs

S2 (Swath)

S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S2_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S2)
A UTC time associated with the scan.
Year (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel2 x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel2 x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
SCstatus (Group in S2)

SCorientation (2-byte integer, array size: nscan2):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan2):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan2):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan2):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan2):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel2 x nscan2):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**

0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-(1-98) = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

**DETAILED SPECIFICATIONS:**

1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

102 = Climatology check warning (19V Channel)
103 = Climatology check warning (19H Channel)
104 = Climatology check warning (22V Channel)
105 = Climatology check warning (37V Channel)
106 = Climatology check warning (37H Channel)
107 = Climatology check warning (19V Channel)
108 = Climatology check warning (19V Channel)
109 = Climatology check warning (Multiple low-res channels)
110 = Climatology check warning (Multiple high-res channels)
111 = Warning adjacent/cross-pol pixel flagged as bad
112 = Warning of increased noise in 85V channel on DMSP F08
113 = RADCAL correction applied to Tb22v (do not use for climate)
114 = Ta correction made by eliminating spikes in scan cal data

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

-102 = Climatology check flagged in input BASE file
-103 = Climatology check failed (19V Channel)
-104 = Climatology check failed (19H Channel)
-105 = Climatology check failed (22V Channel)
-106 = Climatology check failed (37V Channel)
-107 = Climatology check failed (37H Channel)
-108 = Climatology check failed (85V Channel)
-109 = Climatology check failed (85H Channel)
-110 = Climatology check failed (Multiple low-res channels)
-111 = Climatology check failed (Multiple high-res channels)
-112 = Distance between pixels is nonphysical
-115 = Failure of 85V channel on DMSP F08
-116 = Failure of 85V and increased noise in 85H on DMSP F08
-117 = Failure of both 85V and 85H channels on DMSP F08
-118 = Invalid scan time
-119 = Ta set to missing due to bad cal data
-120 = All data set to missing

**incidenceAngle** (4-byte float, array size: nchUIA2 x npixel2 x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA2 x npixel2 x nscan2):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel2 x nscan2):

Index (1 based as in Fortran) of
the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and
2 unique incidence angles, then the dimensions
in Fortran would be:

incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)

The user would do the following to retrieve the angles
for a given channel, pixel, and scan:

i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)

The incidenceAngleIndex is the same for every scan,
but is repeated each scan for the convenience of users
reading the data scan by scan. In addition,
incidenceAngleIndex is located in metadata for the
convenience of users wishing to read this information
from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value
**Tc** (4-byte float, array size: nchannel2 x npixel2 x nscan2):
GPM Common Calibrated Brightness Temperature. The channels are:

85.5 GHz vertically-polarized TBs
85.5 GHz horizontally-polarized TBs

**C Structure Header file:**

```c
#ifndef _TK_1CSSMI_H_
define _TK_1CSSMI_H_

#ifndef _L1CSSMI_S2_
define _L1CSSMI_S2_

typedef struct {
   SCANTIME ScanTime;
   float Latitude[128];
   float Longitude[128];
   SCSTATUS SCstatus;
   signed char Quality[128];
   float incidenceAngle[128][1];
   signed char sunGlintAngle[128][1];
   signed char incidenceAngleIndex[2];
   float Tc[128][2];
} L1CSSMI_S2;
#endif

#endif

#ifndef _SCSTATUS_
define _SCSTATUS_

typedef struct {
   short SCorientation;
   float SClatitude;
   float SClongitude;
   float SCaltitude;
   double FractionalGranuleNumber;
} SCSTATUS;
#endif

#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
   short SCorientation;
   float SClatitude;
   float SClongitude;
   float SCaltitude;
   double FractionalGranuleNumber;
} SCANTIME;
#endif
```
typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L1CSSMI_S1_
#define _L1CSSMI_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[64];
    float Longitude[64];
    SCSTATUS SCstatus;
    signed char Quality[64];
    float incidenceAngle[64][1];
    signed char sunGlintAngle[64][1];
    signed char incidenceAngleIndex[5];
    float Tc[64][5];
} L1CSSMI_S1;

#endif

#ifndef _L1CSSMI_SWATHS_
#define _L1CSSMI_SWATHS_

typedef struct {
    L1CSSMI_S1 S1;
    L1CSSMI_S2 S2[2];
} L1CSSMI_SWATHS;

#endif

#endif
NOTE: $S2[0]$ contains A-scan data and $S2[1]$ contains B-scan data.

**Fortran Structure Header file:**

```
STRUCTURE /L1CSSMI_S2/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(128)
   REAL*4 Longitude(128)
   RECORD /SCSTATUS/ SCstatus
   BYTE Quality(128)
   REAL*4 incidenceAngle(1,128)
   BYTE sunGlintAngle(1,128)
   BYTE incidenceAngleIndex(2)
   REAL*4 Tc(2,128)
END STRUCTURE

STRUCTURE /SCSTATUS/
   INTEGER*2 SCorientation
   REAL*4 SClatitude
   REAL*4 SClongitude
   REAL*4 SCaltitude
   REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
   INTEGER*2 Year
   BYTE Month
   BYTE DayOfMonth
   BYTE Hour
   BYTE Minute
   BYTE Second
   INTEGER*2 MilliSecond
   INTEGER*2 DayOfYear
   REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1CSSMI_S1/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(64)
   REAL*4 Longitude(64)
   RECORD /SCSTATUS/ SCstatus
   BYTE Quality(64)
```
REAL*4 incidenceAngle(1,64)
BYTE sunGlintAngle(1,64)
BYTE incidenceAngleIndex(5)
REAL*4 Tc(5,64)
END STRUCTURE

STRUCTURE /L1CSSMI_SWATHS/
    RECORD /L1CSSMI_S1/ S1;
    RECORD /L1CSSMI_S2/ S2(2);
END STRUCTURE

NOTE: S2(1) contains A-scan data
     and S2(2) contains B-scan data.

5.24 1CSSMIS - Common Calibrated Brightness Temperature

1CSSMIS contains common calibrated brightness temperature from the SSMIS passive microwave instruments flown on the DMSP satellites. Swath S1 has 3 low frequency channels (19V 19H 22V). Swath S2 has 2 low frequency channels (37V 37H). Swath S3 has 4 high frequency channels (150H 183+/−1H 183+/−3H 183+/−7H). S4 has 2 high frequency channels (91V 91H). All the above frequencies are in GHz.

Earth observations for all four swaths are taken during a 144° segment of the instrument rotation when SSMIS scans in the direction of forward satellite motion. We define the spacecraft vector (v) at the center of this segment. "v" is used in the definition of the variable SCorientation.

RELATION BETWEEN THE SWATHS: Each S1 and S2 scan contains low frequency channels sampled 90 times along the scan. Each S3 and S4 scan contains high frequency channels sampled 180 times along the scan. All four swaths have exactly the same number of scans. All four swaths repeat scans every 1.9s. The earth positions of S1 are very close to those of S2. The earth positions of S3 are very close to those of S4. The earth positions of S1 and S2 alternate with those of S3 and S4 along the satellite track. The positions of the S1 and S2 pixels do not match the positions of the S3 and S4 pixels.

The Figure below shows the locations of the samples of Swath S1 and Swath S2 scan 1 and Swath S3 and Swath S4 scan 1. Each "+" represents centers of samples from two swaths. For example, the label "S1S2:1,2" means that Swath S1 and Swath S2, scan 1, sample 2 is located approximately at the "+". The positions of S1 and S2 are slightly different from each other but close enough to be represented by the same "+" in the Figure. The positions of S3 and S4 are slightly different from each other but close enough to be represented by the same "+" in the Figure.

S1S2:1,1  S1S2:1,2  S1S2:1,90
Known problems or issues: L1C data was flagged and Tc was set to Missing due to channel failure: F18 150GHz starting Feb 14, 2012. F16 183GHz starting Dec 1, 2013 F16 150GHz starting May 1, 2015.

Dimension definitions:

- **nscan1**: var Number of Swath S1 scans in the granule.
- **nchannel1**: 3 Number of Swath S1 channels.
- **npixel1**: 90 Number of Swath S1 pixels in one scan.
- **nchUIA1**: 1 Number of Swath S1 unique incidence angles.
- **nscan2**: var Number of Swath S2 scans in the granule.
- **nchannel2**: 2 Number of Swath S2 channels.
- **npixel2**: 90 Number of Swath S2 pixels in one scan.
- **nchUIA2**: 1 Number of Swath S2 unique incidence angles.
- **nscan3**: var Number of Swath S3 scans in the granule.
- **nchannel3**: 4 Number of Swath S3 channels.
- **npixel3**: 180 Number of Swath S3 pixels in one scan.
- **nchUIA3**: 1 Number of Swath S3 unique incidence angles.
- **nscan4**: var Number of Swath S4 scans in the granule.
- **nchannel4**: 2 Number of Swath S4 channels.
- **npixel4**: 180 Number of Swath S4 pixels in one scan.
- **nchUIA4**: 1 Number of Swath S4 unique incidence angles.

Figure 338 through Figure 350 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.
Figure 338: Data Format Structure for 1CSSMIS, Common Calibrated Brightness Temperature

Figure 339: Data Format Structure for 1CSSMIS, S1
Figure 340: Data Format Structure for 1CSSMIS, S2

Figure 341: Data Format Structure for 1CSSMIS, S3
5.24 1CSSMIS - Common Calibrated Brightness Temperature

Figure 342: Data Format Structure for 1CSSMIS, S4

Figure 343: Data Format Structure for 1CSSMIS, S1, ScanTime

Figure 344: Data Format Structure for 1CSSMIS, S1, SCstatus
Figure 345: Data Format Structure for 1CSSMIS, S2, ScanTime

Figure 346: Data Format Structure for 1CSSMIS, S2, SCstatus

Figure 347: Data Format Structure for 1CSSMIS, S3, ScanTime
Figure 348: Data Format Structure for 1CSSMIS, S3, SCstatus

Figure 349: Data Format Structure for 1CSSMIS, S4, ScanTime

Figure 350: Data Format Structure for 1CSSMIS, S4, SCstatus
XCALinfo (Metadata):
XCALinfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

S1 (Swath)

S1_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S1_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S1)
A UTC time associated with the scan.
Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value
**DayOfYear** (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
- 9999  Missing value

**SecondOfDay** (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
- 9999.9  Missing value

**Latitude** (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
- 9999.9  Missing value

**Longitude** (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9  Missing value

**SCstatus** (Group in S1)

**SCorientation** (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
- 9999  Missing value

**SClatitude** (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
- 9999.9  Missing value

**SClongitude** (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9  Missing value

**SCaltitude** (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
- 9999.9  Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**Quality** (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**
- 0 = Good data in all channels in the swath
- gt 0 = Cautionary warning flags
- 1-99 = Generic flags (all sensors)
- 100-127 = Sensor specific flags
- lt 0 = Major errors resulting in missing data
- -(1-98) = Generic flags (all sensors)
- -99 = Missing value (no quality information available)
- -(100-127) = Sensor specific flags

**DETAILED SPECIFICATIONS:**
- 1 = Possible sunGlint, 0 le sunGlintAnglet 20
- 2 = Possible radio frequency interference
- 3 = Degraded geolocation data
- 4 = Data corrected for warm load intrusion
- 102 = Climatology check warning 19V channel
- 103 = Climatology check warning 19H channel
- 104 = Climatology check warning 22V channel
- 105 = Climatology check warning 37V channel
- 106 = Climatology check warning 37H channel
- 107 = Climatology check warning 91V channel
- 108 = Climatology check warning 91H channel
- 109 = Climatology check warning 150H channel
- 110 = Climatology check warning 183+/-1 channel
- 111 = Climatology check warning 183+/-3 channel
- 112 = Climatology check warning 183+/-7 channel
- 113 = Climatology check warning Multiple enviro sensor channels
- 114 = Climatology check warning Multiple imager sensor channels
- 115 = Climatology check warning One or more LAS sensor channels
- 116 = Climatology check warning One or more UAS sensor channels
- 117 = Climatology check warning Correction for lunar intrusion into warm load
- 118 = Climatology check warning Correction for solar intrusion into warm load
- 119 = No sun angle correction warning in multiple channels
- 120 = Sensor data issue warning in multiple imager sensor channels
- 121 = Sensor data issue warning in multiple enviro sensor channels
- 122 = Sensor data issue warning in 91H channel
-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

-102 = Climatology check flagged in input BASE file
-110 = Climatology check failure 19V channel
-111 = Climatology check failure 19H channel
-112 = Climatology check failure 22V channel
-113 = Climatology check failure 37V channel
-114 = Climatology check failure 37H channel
-115 = Climatology check failure 91V channel
-116 = Climatology check failure 91H channel
-117 = Climatology check failure 150H channel
-118 = Climatology check failure 183+/−1 channel
-119 = Climatology check failure 183+/−3 channel
-120 = Climatology check failure 183+/−7 channel
-121 = Climatology check failure Multiple enviro sensor channels
-122 = Climatology check failure Multiple imager sensor channels
-123 = Climatology check failure One or more LAS sensor channels
-124 = Climatology check failure One or more UAS sensor channels
-125 = Failure of 150H channel
-126 = Failure of multiple imager sensor channels
-127 = Failure of 37V channel

incidenceAngle (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel1 x nscan1):
Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

\[
\text{incidenceAngle}(2, \text{npixel}, \text{nscan}) \\
\text{sunGlintAngle}(2, \text{npixel}, \text{nscan}) \\
\text{incidenceAngleIndex}(10, \text{nscan}) \\
\text{Tc}(10, \text{npixel}, \text{nscan})
\]

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
i = \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
i_a = \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
s_ga = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\]

The \text{incidenceAngleIndex} is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, \text{incidenceAngleIndex} is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
- 99 Missing value

\text{Tc} (4-byte float, array size: \text{nchannel1} x \text{npixel1} x \text{nscan1}): GPM Common Calibrated Brightness Temperature. The channels are:

- 19.35 GHz vertically-polarized TBs
- 19.35 GHz horizontally-polarized TBs
- 22.235 GHz vertically-polarized TBs

\text{S2} (Swath)

\text{S2_SwathHeader} (Metadata): SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data data products. See Metadata for GPM Products for details.
**S2_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S2)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npixel2 x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixel2 x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCstatus** (Group in S2)

**SCorientation** (2-byte integer, array size: nscan2):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

**SClatitude** (4-byte float, array size: nscan2):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**SClongitude** (4-byte float, array size: nscan2):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCaltitude** (4-byte float, array size: nscan2):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan2):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**Quality** (1-byte integer, array size: npixel2 x nscan2):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS**:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
     1-99 = Generic flags (all sensors)
     100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
   -(1-98) = Generic flags (all sensors)
   -99 = Missing value (no quality information available)
   -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
   1 = Possible sunGlint, 0 le sunGlintAngle lt 20
   2 = Possible radio frequency interference
   3 = Degraded geolocation data
   4 = Data corrected for warm load intrusion

102 = Climatology check warning 19V channel
103 = Climatology check warning 19H channel
104 = Climatology check warning 22V channel
105 = Climatology check warning 37V channel
106 = Climatology check warning 37H channel
107 = Climatology check warning 91V channel
108 = Climatology check warning 91H channel
109 = Climatology check warning 150H channel
110 = Climatology check warning 183+/-1 channel
111 = Climatology check warning 183+/-3 channel
112 = Climatology check warning 183+/-7 channel
113 = Climatology check warning Multiple enviro sensor channels
114 = Climatology check warning Multiple imager sensor channels
115 = Climatology check warning One or more LAS sensor channels
116 = Climatology check warning One or more UAS sensor channels
117 = Climatology check warning Correction for lunar intrusion into warm load
118 = Climatology check warning Correction for solar intrusion into warm load
119 = No sun angle correction warning in multiple channels
120 = Sensor data issue warning in multiple imager sensor channels
121 = Sensor data issue warning in multiple enviro sensor channels
122 = Sensor data issue warning in 91H channel

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)
-102 = Climatology check flagged in input BASE file
-110 = Climatology check failure 19V channel
-111 = Climatology check failure 19H channel
-112 = Climatology check failure 22V channel
-113 = Climatology check failure 37V channel
-114 = Climatology check failure 37H channel
-115 = Climatology check failure 91V channel
-116 = Climatology check failure 91H channel
-117 = Climatology check failure 150H channel
-118 = Climatology check failure 183+/-1 channel
-119 = Climatology check failure 183+/-3 channel
-120 = Climatology check failure 183+/-7 channel
-121 = Climatology check failure Multiple enviro sensor channels
-122 = Climatology check failure Multiple imager sensor channels
-123 = Climatology check failure One or more LAS sensor channels
-124 = Climatology check failure One or more UAS sensor channels
-125 = Failure of 150H channel
-126 = Failure of multiple imager sensor channels
-127 = Failure of 37V channel

incidenceAngle (4-byte float, array size: nchUIA2 x npixel2 x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUIA2 x npixel2 x nscan2):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel2 x nscan2):
Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
\begin{align*}
i &= \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
\text{ia} &= \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
\text{sga} &= \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\end{align*}
\]

The `incidenceAngleIndex` is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, `incidenceAngleIndex` is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

\textbf{Tc} (4-byte float, array size: nchannel2 x npixel2 x nscan2):
GPM Common Calibrated Brightness Temperature. The channels are:

\begin{align*}
37.0 & \text{ GHz vertically-polarized TBs} \\
37.0 & \text{ GHz horizontally-polarized TBs}
\end{align*}

\textbf{S3} (Swath)

\textbf{S3\_SwathHeader} (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

\textbf{S3\_IncidenceAngleIndex} (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

\textbf{ScanTime} (Group in S3)
A UTC time associated with the scan.

\textbf{Year} (2-byte integer, array size: nscan3):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined
-9999  Missing value

**Month** (1-byte integer, array size: nscan3):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

**DayOfMonth** (1-byte integer, array size: nscan3):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

**Hour** (1-byte integer, array size: nscan3):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

**Minute** (1-byte integer, array size: nscan3):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

**Second** (1-byte integer, array size: nscan3):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

**MilliSecond** (2-byte integer, array size: nscan3):
Thousands of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

**DayOfYear** (2-byte integer, array size: nscan3):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

**SecondOfDay** (8-byte float, array size: nscan3):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

**Latitude** (4-byte float, array size: npixel3 x nscan3):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**Longitude** (4-byte float, array size: npixel3 x nscan3):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value
SCstatus (Group in S3)

SCorientation (2-byte integer, array size: nscan3):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999  Missing value

SClatitude (4-byte float, array size: nscan3):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

SClongitude (4-byte float, array size: nscan3):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCaltitude (4-byte float, array size: nscan3):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9  Missing value

FractionalGranuleNumber (8-byte float, array size: nscan3):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

Quality (1-byte integer, array size: npixel3 x nscan3):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
  0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
    1-99 = Generic flags (all sensors)
    100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
    -1-98 = Generic flags (all sensors)
    -99 = Missing value (no quality information available)
    -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
  1 = Possible sunGlint, 0 le sunGlintAngle lt 20
  2 = Possible radio frequency interference
  3 = Degraded geolocation data
  4 = Data corrected for warm load intrusion
102 = Climatology check warning 19V channel
103 = Climatology check warning 19H channel
104 = Climatology check warning 22V channel
105 = Climatology check warning 37V channel
106 = Climatology check warning 37H channel
107 = Climatology check warning 91V channel
108 = Climatology check warning 91H channel
109 = Climatology check warning 150H channel
110 = Climatology check warning 183+/-1 channel
111 = Climatology check warning 183+/-3 channel
112 = Climatology check warning 183+/-7 channel
113 = Climatology check warning Multiple enviro sensor channels
114 = Climatology check warning Multiple imager sensor channels
115 = Climatology check warning One or more LAS sensor channels
116 = Climatology check warning One or more UAS sensor channels
117 = Climatology check warning Correction for lunar intrusion into warm load
118 = Climatology check warning Correction for solar intrusion into warm load
119 = No sun angle correction warning in multiple channels
120 = Sensor data issue warning in multiple imager sensor channels
121 = Sensor data issue warning in multiple enviro sensor channels
122 = Sensor data issue warning in 91H channel

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

-102 = Climatology check flagged in input BASE file
-110 = Climatology check failure 19V channel
-111 = Climatology check failure 19H channel
-112 = Climatology check failure 22V channel
-113 = Climatology check failure 37V channel
-114 = Climatology check failure 37H channel
-115 = Climatology check failure 91V channel
-116 = Climatology check failure 91H channel
-117 = Climatology check failure 150H channel
-118 = Climatology check failure 183+/-1 channel
-119 = Climatology check failure 183+/-3 channel
-120 = Climatology check failure 183+/-7 channel
-121 = Climatology check failure Multiple enviro sensor channels
-122 = Climatology check failure Multiple imager sensor channels
-123 = Climatology check failure One or more LAS sensor channels
-124 = Climatology check failure One or more UAS sensor channels
-125 = Failure of 150H channel
-126 = Failure of multiple imager sensor channels
-127 = Failure of 37V channel

**incidenceAngle** (4-byte float, array size: nchUIA3 x npixel3 x nscan3):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA3 x npixel3 x nscan3):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel3 x nscan3):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

```
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
```

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```
i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)
```

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users.
reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
- **99** Missing value

**Tc** (4-byte float, array size: nchannel3 x npixel3 x nscan3):
GPM Common Calibrated Brightness Temperature. The channels are:

- 150 GHz horizontally-polarized TBs
- 183.31 +/- 1 GHz horizontally-polarized TBs
- 183.31 +/- 3 GHz horizontally-polarized TBs
- 183.31 +/- 6.6 GHz horizontally-polarized TBs

**S4** (Swath)

**S4_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S4_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S4)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan4):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- **9999** Missing value

**Month** (1-byte integer, array size: nscan4):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
- **99** Missing value

**DayOfMonth** (1-byte integer, array size: nscan4):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
- **99** Missing value
Hour (1-byte integer, array size: nscan4):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan4):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan4):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan4):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan4):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan4):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel4 x nscan4):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel4 x nscan4):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group in S4)

SCorientation (2-byte integer, array size: nscan4):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value
SClatitude (4-byte float, array size: nscan4):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan4):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan4):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan4):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel4 x nscan4):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
  1-99 = Generic flags (all sensors)
  100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
  -(1-98) = Generic flags (all sensors)
     -99 = Missing value (no quality information available)
     -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

102 = Climatology check warning 19V channel
103 = Climatology check warning 19H channel
104 = Climatology check warning 22V channel
105 = Climatology check warning 37V channel
106 = Climatology check warning 37H channel
107 = Climatology check warning 91V channel
108 = Climatology check warning 91H channel
109 = Climatology check warning 150H channel
110 = Climatology check warning 183+/-1 channel
111 = Climatology check warning 183+/-3 channel
112 = Climatology check warning 183+/-7 channel
113 = Climatology check warning Multiple enviro sensor channels
114 = Climatology check warning Multiple imager sensor channels
115 = Climatology check warning One or more LAS sensor channels
116 = Climatology check warning One or more UAS sensor channels
117 = Climatology check warning Correction for lunar intrusion into warm load
118 = Climatology check warning Correction for solar intrusion into warm load
119 = No sun angle correction warning in multiple channels
120 = Sensor data issue warning in multiple imager sensor channels
121 = Sensor data issue warning in multiple enviro sensor channels
122 = Sensor data issue warning in 91H channel

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km
     threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

-102 = Climatology check flagged in input BASE file
-110 = Climatology check failure 19V channel
-111 = Climatology check failure 19H channel
-112 = Climatology check failure 22V channel
-113 = Climatology check failure 37V channel
-114 = Climatology check failure 37H channel
-115 = Climatology check failure 91V channel
-116 = Climatology check failure 91H channel
-117 = Climatology check failure 150H channel
-118 = Climatology check failure 183+/-1 channel
-119 = Climatology check failure 183+/-3 channel
-120 = Climatology check failure 183+/-7 channel
-121 = Climatology check failure Multiple enviro sensor channels
-122 = Climatology check failure Multiple imager sensor channels
-123 = Climatology check failure One or more LAS sensor channels
-124 = Climatology check failure One or more UAS sensor channels
-125 = Failure of 150H channel
-126 = Failure of multiple imager sensor channels
-127 = Failure of 37V channel

**incidenceAngle** (4-byte float, array size: nchUIA4 x npixel4 x nscan4):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA4 x npixel4 x nscan4):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel4 x nscan4):
Index (1 based as in Fortran) of the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

```
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
```

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```
i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)
```

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

**Values range from 0 to 100. Special values are defined as:**
-99 Missing value
Tc (4-byte float, array size: nchannel4 x npixel4 x nscan4):
GPM Common Calibrated Brightness Temperature. The channels are:

- 91.665 GHz vertically-polarized TBs
- 91.665 GHz horizontally-polarized TBs

C Structure Header file:

```c
#ifndef _TK_1CSSMIS_H_
#define _TK_1CSSMIS_H_

#ifndef _L1CSSMIS_S4_
#define _L1CSSMIS_S4_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[180];
    float Longitude[180];
    SCSTATUS SCstatus;
    signed char Quality[180];
    float incidenceAngle[180][1];
    signed char sunGlintAngle[180][1];
    signed char incidenceAngleIndex[2];
    float Tc[180][2];
} L1CSSMIS_S4;
#endif

#undef _L1CSSMIS_S4_
#define _L1CSSMIS_S3_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[180];
    float Longitude[180];
    SCSTATUS SCstatus;
    signed char Quality[180];
    float incidenceAngle[180][1];
    signed char sunGlintAngle[180][1];
    signed char incidenceAngleIndex[4];
    float Tc[180][4];
} L1CSSMIS_S3;
```

#ifndef _L1CSSMIS_S2_
#define _L1CSSMIS_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[90];
    float Longitude[90];
    SCSTATUS SCstatus;
    signed char Quality[90];
    float incidenceAngle[90][1];
    signed char sunGlintAngle[90][1];
    signed char incidenceAngleIndex[2];
    float Tc[90][2];
} L1CSSMIS_S2;

#endif

#ifndef _SCSTATUS_
#define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    float SClatitude;
    float SClongitude;
    double FractionalGranuleNumber;
} SCSTATUS;

#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
} SCANTIME;

#endif
typedef struct {
  SCANTIME ScanTime;
  float Latitude[90];
  float Longitude[90];
  SCSTATUS SCstatus;
  signed char Quality[90];
  float incidenceAngle[90][1];
  signed char sunGlintAngle[90][1];
  signed char incidenceAngleIndex[3];
  float Tc[90][3];
} L1CSSMIS_S1;

typedef struct {
  L1CSSMIS_S1 S1;
  L1CSSMIS_S2 S2;
  L1CSSMIS_S3 S3;
  L1CSSMIS_S4 S4;
} L1CSSMIS_SWATHS;

Fortran Structure Header file:

STRUCTURE /L1CSSMIS_S4/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(180)
  REAL*4 Longitude(180)
RECORD /SCSTATUS/ SCstatus
BYTE Quality(180)
REAL*4 incidenceAngle(1,180)
BYTE sunGlintAngle(1,180)
BYTE incidenceAngleIndex(2)
REAL*4 Tc(2,180)
END STRUCTURE

STRUCTURE /L1CSSMIS_S3/
RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(180)
REAL*4 Longitude(180)
RECORD /SCSTATUS/ SCstatus
BYTE Quality(180)
REAL*4 incidenceAngle(1,180)
BYTE sunGlintAngle(1,180)
BYTE incidenceAngleIndex(4)
REAL*4 Tc(4,180)
END STRUCTURE

STRUCTURE /L1CSSMIS_S2/
RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(90)
REAL*4 Longitude(90)
RECORD /SCSTATUS/ SCstatus
BYTE Quality(90)
REAL*4 incidenceAngle(1,90)
BYTE sunGlintAngle(1,90)
BYTE incidenceAngleIndex(2)
REAL*4 Tc(2,90)
END STRUCTURE

STRUCTURE /SCSTATUS/
INTEGER*2 SCorientation
REAL*4 SClatitude
REAL*4 SClongitude
REAL*4 SCaltitude
REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
INTEGER*2 Year
BYTE Month
5.25 1CAMSRE - Common Calibrated Brightness Temperature

1CAMSRE contains common calibrated brightness temperature from the AMSR-E passive microwave instrument flown on the AQUA satellite. This product contains 6 swaths. Swath 1 has channels 10.65V 10.65H. Swath 2 has channels 18.7V 18.7H. Swath 3 has channels 23.8V 23.8H. Swath 4 has channels 36.5V 36.5H. Swath S5 has 2 high frequency A-Scan channels (89V 89H). Swath S6 has 2 high frequency B-Scan channels (89V 89H). Data for all six swaths is observed in the same revolution of the instrument. High frequency A-Scan and high frequency B-Scan data are observed in separate feedhorns.

Earth observations for all three swaths are taken during a 122° segment of the instrument rotation when AMSR-E is looking in the forward direction. We define the spacecraft vector (v) at the center of this segment. "v" is used in the definition of the variable SCorientation.
RELATION BETWEEN THE SWATHS: Each S1 scan contains 10 GHz channels sampled 196 times along the scan. S2, S3, and S4 are sampled nominally at the same position as the S1 samples, but differ by small distances. Each S5 scan contains high frequency A channels sampled 392 times along the scan. Each S6 scan contains high frequency B channels sampled 392 times along the scan. Both Swath S5 and Swath S6 have exactly twice as many pixels as Swath S1. S1 pixels 1, 2, 3, ... coincide with S5 pixels 1, 3, 5, ... Scans of all swaths are repeated every 1.5s and the scans of one swath are about 10km apart along the direction of the satellite track. Along an S1 scan every other center of an S5 pixel coincides with the center of an S1 pixel, but the S6 pixels are offset from S1 and S5 pixels by nominally 15km in the direction normal to the scan direction on the aft side, in other words S6 pixels are nominally 15km "behind" the S1 and S5 pixels for the same scan.

The Figure below shows the locations of the pixels of scans 1 and 2 for swaths S1, S5, and S6. Since swaths S2, S3 and S4 are close to S1, they are omitted from the figure. Each "+" represents centers of pixels from one or more swaths. For example, the label "S1:1,2 S5:1,3" means that both Swath S1, Scan 1, Pixel 2 and Swath S5, Scan 1, Pixel 3 are located at the "+".
S1:1,1 S5:1,1 S1:1,2 S5:1,3 S1:1,196 S5:1,391 S5:1,392
+ + + + + + + + + +

S1:2,1 S5:2,1 S1:2,2 S5:2,3 S1:2,196 S5:2,391 S5:2,392
+ + + + + + + + + +

KNOWN PROBLEMS OR ISSUES:
1. Swath S5 (89A) V and H data is missing due to channel failure starting Nov. 4, 2004

Dimension definitions:
- nscan1 var Number of scans in Swath S1 in the granule.
- nscan2 var Number of scans in Swath S2 in the granule.
- nscan3 var Number of scans in Swath S3 in the granule.
- nscan4 var Number of scans in Swath S4 in the granule.
- nscan5 var Number of scans in Swath S5 in the granule.
- nscan6 var Number of scans in Swath S6 in the granule.
- npixel1 196 Number of Swath S1 pixels in one scan.
- npixel2 196 Number of Swath S2 pixels in one scan.
- npixel3 196 Number of Swath S3 pixels in one scan.
- npixel4 196 Number of Swath S4 pixels in one scan.
- npixel5 392 Number of Swath S5 pixels in one scan.
- npixel6 392 Number of Swath S6 pixels in one scan.
- nchannel1 2 Number of Swath S1 channels.
- nchannel2 2 Number of Swath S2 channels.
- nchannel3 2 Number of Swath S3 channels.
- nchannel4 2 Number of Swath S4 channels.
- nchannel5 2 Number of Swath S5 channels.
- nchannel6 2 Number of Swath S6 channels.
- nchUIA1 1 Number of Swath S1 unique incidence angles.
- nchUIA2 1 Number of Swath S2 unique incidence angles.
- nchUIA3 1 Number of Swath S3 unique incidence angles.
- nchUIA4 1 Number of Swath S4 unique incidence angles.
- nchUIA5 1 Number of Swath S5 unique incidence angles.
- nchUIA6 1 Number of Swath S6 unique incidence angles.

Figure 351 through Figure 369 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 351: Data Format Structure for 1CAMSRE, Common Calibrated Brightness Temperature

Figure 352: Data Format Structure for 1CAMSRE, S1
Figure 353: Data Format Structure for 1CAMSRE, S2

Figure 354: Data Format Structure for 1CAMSRE, S3
Figure 355: Data Format Structure for 1CAMSRE, S4

Figure 356: Data Format Structure for 1CAMSRE, S5
5.25 1CAMSRE - Common Calibrated Brightness Temperature

Figure 357: Data Format Structure for 1CAMSRE, S6

Figure 358: Data Format Structure for 1CAMSRE, S1, ScanTime

Figure 359: Data Format Structure for 1CAMSRE, S1, SCstatus
Figure 360: Data Format Structure for 1CAMSRE, S2, ScanTime

Figure 361: Data Format Structure for 1CAMSRE, S2, SCstatus

Figure 362: Data Format Structure for 1CAMSRE, S3, ScanTime
Figure 363: Data Format Structure for 1CAMSRE, S3, SCstatus

Figure 364: Data Format Structure for 1CAMSRE, S4, ScanTime

Figure 365: Data Format Structure for 1CAMSRE, S4, SCstatus
Figure 366: Data Format Structure for 1CAMSRE, S5, ScanTime
5.25 1CAMSRE - Common Calibrated Brightness Temperature

**SCstatus**
- SCorientation 2 bytes Array: nscan5
- SClatitude 4 bytes Array: nscan5
- SClongitude 4 bytes Array: nscan5
- SCalitude 4 bytes Array: nscan5
- FractionalGranuleNumber 8 bytes Array: nscan5

Figure 367: Data Format Structure for 1CAMSRE, S5, SCstatus

**ScanTime**
- Year 2 bytes Array: nscan6
- Month 1 byte Array: nscan6
- DayOfMonth 1 byte Array: nscan6
- Hour 1 byte Array: nscan6
- Minute 1 byte Array: nscan6
- Second 1 byte Array: nscan6
- MilliSecond 2 bytes Array: nscan6
- DayOfYear 2 bytes Array: nscan6
- SecondOfDay 8 bytes Array: nscan6

Figure 368: Data Format Structure for 1CAMSRE, S6, ScanTime

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**SCstatus**
- SCorientation 2 bytes Array: nscan6
- SClatitude 4 bytes Array: nscan6
- SClongitude 4 bytes Array: nscan6
- SCalitude 4 bytes Array: nscan6
- FractionalGranuleNumber 8 bytes Array: nscan6

Figure 369: Data Format Structure for 1CAMSRE, S6, SCstatus
NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

XCALinfo (Metadata):
XCALinfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

S1 (Swath)

S1_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S1_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S1)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
   -9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
   -99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
   -99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
   -99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
   -99 Missing value
**Second** (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan1):
Thousands of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npixel1 x nscan1):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

**Longitude** (4-byte float, array size: npixel1 x nscan1):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

**SCstatus** (Group in S1)

**SCorientation** (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

**SClatitude** (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**SClongitude** (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCaltitude** (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**Quality** (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-(1-98) = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

**DETAILED SPECIFICATIONS:**
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

-100 = Failure of 89A V/H channel on AMSRE Nov 04, 2004

**incidenceAngle** (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

- incidenceAngle(2,npixel,nscan)
- sunGlintAngle(2,npixel,nscan)
- incidenceAngleIndex(10,nscan)
- Tc(10,npixel,nscan)

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```
i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)
```

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:

-99  Missing value

**Tc** (4-byte float, array size: nchannel1 x npixel1 x nscan1):

GPM Common Calibrated Brightness Temperature. The channels are:

- 10.65 GHz vertically-polarized TBs
- 10.65 GHz horizontally-polarized TBs

**S2** (Swath)
S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S2_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value
**Latitude** (4-byte float, array size: npixel1 x nscan1):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

**Longitude** (4-byte float, array size: npixel1 x nscan1):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

**SCstatus** (Group in S2)

**SCorientation** (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

**SClatitude** (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**SClongitude** (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCaltitude** (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**Quality** (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**
- 0 = Good data in all channels in the swath
- > 0 = Cautionary warning flags
  - 1-99 = Generic flags (all sensors)
  - 100-127 = Sensor specific flags
- < 0 = Major errors resulting in missing data
-(1-98) = Generic flags (all sensors)
   -99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km
threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

**incidenceAngle** (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
i = \text{incidenceAngleIndex}(\text{channel}, \text{scan})
\]
\[
ia = \text{incidenceAngle}(i, \text{pixel}, \text{scan})
\]
\[
sga = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99   Missing value

\(T_c\) (4-byte float, array size: nchannel x npixel x nscan):
GPM Common Calibrated Brightness Temperature. The channels are:

18.7  GHz vertically-polarized TBs
18.7  GHz horizontally-polarized TBs

S3 (Swath)

\(S3_{-SwathHeader}\) (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

\(S3_{-IncidenceAngleIndex}\) (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S3)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999   Missing value
Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
Nominal latitude of the observation point on the earth surface at low frequency. This was
calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

Longitude (4-byte float, array size: npixel1 x nscan1):
Nominal longitude of the observation point on the earth surface at low frequency. This was
calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

SCstatus (Group in S3)

SCorientation (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. The relationship of v to the sensor geometry is defined
in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values
are defined as:
   -9999    Missing value

**SCLatitude** (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
   -9999.9    Missing value

**SCLongitude** (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9    Missing value

**SCAltitude** (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
   -9999.9    Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
   -9999.9    Missing value

**Quality** (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**
   0 = Good data in all channels in the swath
   gt 0 = Cautionary warning flags
   1-99 = Generic flags (all sensors)
   100-127 = Sensor specific flags
   lt 0 = Major errors resulting in missing data
   -(1-98) = Generic flags (all sensors)
   -99 = Missing value (no quality information available)
   -(100-127) = Sensor specific flags

**DETAILED SPECIFICATIONS:**
  1 = Possible sunGlint, 0 le sunGlintAngle lt 20
  2 = Possible radio frequency interference
  3 = Degraded geolocation data
  4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

**incidenceAngle** (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
i = \text{incidenceAngleIndex}(\text{channel},\text{scan})
\]
\[
\text{ia} = \text{incidenceAngle}(i,\text{pixel},\text{scan})
\]
\[
\text{sga} = \text{sunGlintAngle}(i,\text{pixel},\text{scan})
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.
Values range from 0 to 100. Special values are defined as:
-99  Missing value

**Tc** (4-byte float, array size: nchannel1 x npixel1 x nscan1):
GPM Common Calibrated Brightness Temperature. The channels are:

23.8 GHz vertically-polarized TBs
23.8 GHz horizontally-polarized TBs

**S4** (Swath)

**S4_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S4_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S4)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

**DayOfMonth** (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

**Hour** (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

**Minute** (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value
Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
Nominal latitude of the observation point on the earth surface at low frequency. This was
calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

Longitude (4-byte float, array size: npixel1 x nscan1):
Nominal longitude of the observation point on the earth surface at low frequency. This was
calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

SCstatus (Group in S4)

SCorientation (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. The relationship of v to the sensor geometry is defined
in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values
are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**Quality** (1-byte integer, array size: npixel1 x nscan1): Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**
- 0 = Good data in all channels in the swath
- gt 0 = Cautionary warning flags
  - 1-99 = Generic flags (all sensors)
  - 100-127 = Sensor specific flags
- lt 0 = Major errors resulting in missing data
  - -(1-98) = Generic flags (all sensors)
  - -99 = Missing value (no quality information available)
  - -(100-127) = Sensor specific flags

**DETAILED SPECIFICATIONS:**
- 1 = Possible sunGlint, 0 ≤ sunGlintAngle < 20
- 2 = Possible radio frequency interference
- 3 = Degraded geolocation data
- 4 = Data corrected for warm load intrusion
- -1 = Data is missing from file or unreadable, missing scan
- -2 = Invalid Tb or unphysical brightness temperature Tb ≤ 50 or Tb ≥ 350
- -3 = Error in geolocation
- -4 = Data is missing in 1 channel
- -5 = Data is missing in multiple channels
- -6 = Lat/Lon values are out of range
- -7 = Non-normal status modes
- -10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
- -99 = Missing value (no quality information available)

**incidenceAngle** (4-byte float, array size: nchUIA1 x npixel1 x nscan1): Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA1 x npixel1 x nscan1): Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.
incidenceAngleIndex (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

\[
\text{incidenceAngle}(2, npixel, nscan) \\
\text{sunGlintAngle}(2, npixel, nscan) \\
\text{incidenceAngleIndex}(10, nscan) \\
\text{Tc}(10, npixel, nscan)
\]

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
i = \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
i_a = \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
s_g = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

Tc (4-byte float, array size: nchannel1 x npixel1 x nscan1):
GPM Common Calibrated Brightness Temperature. The channels are:

36.5 GHz vertically-polarized TBs
36.5 GHz horizontally-polarized TBs

S5 (Swath)
5.25 1CAMSRE - Common Calibrated Brightness Temperature

**S5_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S5_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S5)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan5):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan5):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan5):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan5):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan5):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan5):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan5):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan5):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan5):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value
Latitude (4-byte float, array size: npixel5 x nscan5):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: npixel5 x nscan5):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCstatus (Group in S5)

SCorientation (2-byte integer, array size: nscan5):
The angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. The relationship of v to the sensor geometry is defined
in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values
are defined as:
-9999  Missing value

SClatitude (4-byte float, array size: nscan5):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

SClongitude (4-byte float, array size: nscan5):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCaltitude (4-byte float, array size: nscan5):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9  Missing value

FractionalGranuleNumber (8-byte float, array size: nscan5):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

Quality (1-byte integer, array size: npixel5 x nscan5):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0  = Good data in all channels in the swath
gt 0 = Cautionary warning flags
   1-99 = Generic flags (all sensors)
   100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
   -(1-98) = Generic flags (all sensors)
   -99 = Missing value (no quality information available)
   -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion
-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km
      threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

incidenceAngle (4-byte float, array size: nchUA5 x npixel5 x nscan5):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
   -9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUA5 x npixel5 x nscan5):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel5 x nscan5):
Index (1 based as in Fortran) of the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:
   incidenceAngle(2,npixel,nscan)
sunGlintAngle(2, npixel, nscan)
incidenceAngleIndex(10, nscan)
Tc(10, npixel, nscan)

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[ i = \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \]
\[ i_a = \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \]
\[ sga = \text{sunGlintAngle}(i, \text{pixel}, \text{scan}) \]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

\( Tc \) (4-byte float, array size: nchannel5 x npixel5 x nscan5):
GPM Common Calibrated Brightness Temperature. The channels are:

89 GHz vertically-polarized TBs
89 GHz horizontally-polarized TBs

**S6** (Swath)

**S6_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S6_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S6)
A UTC time associated with the scan.
5.25 1CAMSRE - Common Calibrated Brightness Temperature

**Year** (2-byte integer, array size: nscan6):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- **9999** Missing value

**Month** (1-byte integer, array size: nscan6):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
- **99** Missing value

**DayOfMonth** (1-byte integer, array size: nscan6):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
- **99** Missing value

**Hour** (1-byte integer, array size: nscan6):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
- **99** Missing value

**Minute** (1-byte integer, array size: nscan6):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
- **99** Missing value

**Second** (1-byte integer, array size: nscan6):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
- **99** Missing value

**MilliSecond** (2-byte integer, array size: nscan6):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
- **9999** Missing value

**DayOfYear** (2-byte integer, array size: nscan6):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
- **9999** Missing value

**SecondOfDay** (8-byte float, array size: nscan6):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
- **9999.9** Missing value

**Latitude** (4-byte float, array size: npixel6 x nscan6):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
- **9999.9** Missing value

**Longitude** (4-byte float, array size: npixel6 x nscan6):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
- **9999.9** Missing value
SCstatus (Group in S6)

SCOrientation (2-byte integer, array size: nscan6):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan6):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan6):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan6):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan6):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel6 x nscan6):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-(1-98) = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

**incidenceAngle** (4-byte float, array size: nchUIA6 x npixel6 x nscan6):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 = Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA6 x npixel6 x nscan6):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel6 x nscan6):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

```fortran
  incidenceAngle(2,npixel,nscan)
  sunGlintAngle(2,npixel,nscan)
  incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
```

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```fortran
i = incidenceAngleIndex(channel,scan)
i = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)
```
The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

Tc (4-byte float, array size: nchannel6 x npixel6 x nscan6):
GPM Common Calibrated Brightness Temperature. The channels are:

89 GHz vertically-polarized TBs
89 GHz horizontally-polarized TBs

C Structure Header file:

```c
#ifndef _TK_1CAMSRE_H_
define _TK_1CAMSRE_H_

#ifndef _L1CAMSRE_S6_
define _L1CAMSRE_S6_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[392];
    float Longitude[392];
    SCSTATUS SCstatus;
    signed char Quality[392];
    float incidenceAngle[392][1];
    signed char sunGlintAngle[392][1];
    signed char incidenceAngleIndex[2];
    float Tc[392][2];
} L1CAMSRE_S6;

#endif

#ifndef _L1CAMSRE_S5_
define _L1CAMSRE_S5_

typedef struct {
```
```
typedef struct {
    SCANTIME ScanTime;
    float Latitude[196];
    float Longitude[196];
    SCSTATUS SCstatus;
    signed char Quality[196];
    float incidenceAngle[196][1];
    signed char sunGlintAngle[196][1];
    signed char incidenceAngleIndex[2];
    float Tc[196][2];
} L1CAMSRE_S4;

#endif
#endif

typedef struct {
    SCANTIME ScanTime;
    float Latitude[196];
    float Longitude[196];
    SCSTATUS SCstatus;
    signed char Quality[196];
    float incidenceAngle[196][1];
    signed char sunGlintAngle[196][1];
    signed char incidenceAngleIndex[2];
    float Tc[196][2];
} L1CAMSRE_S3;

#endif
#endif

typedef struct {
    SCANTIME ScanTime;
    float Latitude[392];
    float Longitude[392];
    SCSTATUS SCstatus;
    signed char Quality[392];
    float incidenceAngle[392][1];
    signed char sunGlintAngle[392][1];
    signed char incidenceAngleIndex[2];
    float Tc[392][2];
} L1CAMSRE_S5;
typedef struct {
    SCANTIME ScanTime;
    float Latitude[196];
    float Longitude[196];
    SCSTATUS SCstatus;
    signed char Quality[196];
    float incidenceAngle[196][1];
    signed char sunGlintAngle[196][1];
    signed char incidenceAngleIndex[2];
    float Tc[196][2];
} L1CAMSRE_S2;

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCalatitude;
    double FractionalGranuleNumber;
} SCSTATUS;

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L1CAMSRE_S1_
#define _L1CAMSRE_S1_
typedef struct {
    SCANTIME ScanTime;
    float Latitude[196];
    float Longitude[196];
    SCSTATUS SCstatus;
    signed char Quality[196];
    float incidenceAngle[196][1];
    signed char sunGlintAngle[196][1];
    signed char incidenceAngleIndex[2];
    float Tc[196][2];
} L1CAMSRE_S1;
#endif

#ifndef _L1CAMSRE_SWATHS_
#define _L1CAMSRE_SWATHS_
typedef struct {
    L1CAMSRE_S1 S1;
    L1CAMSRE_S2 S2;
    L1CAMSRE_S3 S3;
    L1CAMSRE_S4 S4;
    L1CAMSRE_S5 S5;
    L1CAMSRE_S6 S6;
} L1CAMSRE_SWATHS;
#endif

#endif

Fortran Structure Header file:
STRUCTURE /L1CAMSRE_S6/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(392)
  REAL*4 Longitude(392)
  RECORD /SCSTATUS/ SCstatus
  BYTE Quality(392)
  REAL*4 incidenceAngle(1,392)
  BYTE sunGlintAngle(1,392)
  BYTE incidenceAngleIndex(2)
  REAL*4 Tc(2,392)
END STRUCTURE

STRUCTURE /L1CAMSRE_S5/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(392)
  REAL*4 Longitude(392)
  RECORD /SCSTATUS/ SCstatus
  BYTE Quality(392)
  REAL*4 incidenceAngle(1,392)
  BYTE sunGlintAngle(1,392)
  BYTE incidenceAngleIndex(2)
  REAL*4 Tc(2,392)
END STRUCTURE

STRUCTURE /L1CAMSRE_S4/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(196)
  REAL*4 Longitude(196)
  RECORD /SCSTATUS/ SCstatus
  BYTE Quality(196)
  REAL*4 incidenceAngle(1,196)
  BYTE sunGlintAngle(1,196)
  BYTE incidenceAngleIndex(2)
  REAL*4 Tc(2,196)
END STRUCTURE

STRUCTURE /L1CAMSRE_S3/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(196)
  REAL*4 Longitude(196)
  RECORD /SCSTATUS/ SCstatus
  BYTE Quality(196)
  REAL*4 incidenceAngle(1,196)
BYTE sunGlintAngle(1,196)
BYTE incidenceAngleIndex(2)
REAL*4 Tc(2,196)

END STRUCTURE

STRUCTURE /L1CAMSRE_S2/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(196)
    REAL*4 Longitude(196)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(196)
    REAL*4 incidenceAngle(1,196)
    BYTE sunGlintAngle(1,196)
    BYTE incidenceAngleIndex(2)
    REAL*4 Tc(2,196)

END STRUCTURE

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude
    REAL*4 SCaltitude
    REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay

END STRUCTURE

STRUCTURE /L1CAMSRE_S1/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(196)
    REAL*4 Longitude(196)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(196)

END STRUCTURE
REAL*4 incidenceAngle(1,196)
BYTE sunGlintAngle(1,196)
BYTE incidenceAngleIndex(2)
REAL*4 Tc(2,196)
END STRUCTURE

STRUCTURE /L1CAMSRE_SWATHS/
  RECORD /L1CAMSRE_S1/ S1;
  RECORD /L1CAMSRE_S2/ S2;
  RECORD /L1CAMSRE_S3/ S3;
  RECORD /L1CAMSRE_S4/ S4;
  RECORD /L1CAMSRE_S5/ S5;
  RECORD /L1CAMSRE_S6/ S6;
END STRUCTURE

5.26 1CAMSR2 - Common Calibrated Brightness Temperature

1CAMSR2 contains common calibrated brightness temperature from the AMSR2 passive microwave instrument flown on the GCOM-W1 satellite. This product contains 6 swaths. Swath 1 has channels 10.65V 10.65H. Swath 2 has channels 18.7V 18.7H. Swath 3 has channels 23.8V 23.8H. Swath 4 has channels 36.5V 36.5H. Swath S5 has 2 high frequency A-Scan channels (89V 89H). Swath S6 has 2 high frequency B-Scan channels (89V 89H). Data for all six swaths is observed in the same revolution of the instrument. High frequency A and high frequency B data are observed in separate feedhorns.

RELATION BETWEEN THE SWATHS: Each S1 scan contains 10 GHz channels sampled 243 times along the scan. S2, S3, and S4 are sampled nominally at the same position as the S1 samples, but differ by small distances. Each S5 scan contains high frequency A channels sampled 486 times along the scan. Each S6 scan contains high frequency B channels sampled 486 times along the scan. Both Swath S5 and Swath S6 have exactly twice as many pixels as Swath S1. S1 pixels 1, 2, 3, ... coincide with S5 pixels 1, 3, 5, ... Scans of all swaths are repeated every 1.5s and the scans of one swath are about 10km apart along the direction of the satellite track. Along an S1 scan every other center of an S5 pixel coincides with the center of an S1 pixel, but the S6 pixels are offset from S1 and S5 pixels by nominally 15km in the direction normal to the scan direction on the aft side, in other words S6 pixels are nominally 15km "behind" the S1 and S5 pixels for the same scan.

The Figure below shows the locations of the pixels of scans 1 and 2 for swaths S1, S5, and S6. Since swaths S2, S3 and S4 are close to S1, they are omitted from the figure. Each "+" represents centers of pixels from one or more swaths. For example, the label "S1:1,2 S5:1,3" means that both Swath S1, Scan 1, Pixel 2 and Swath S5, Scan 1, Pixel 3 are located at the "+".
<table>
<thead>
<tr>
<th>Dimension</th>
<th>Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>S6:1,1</td>
<td></td>
</tr>
<tr>
<td>S6:1,2</td>
<td></td>
</tr>
<tr>
<td>S6:1,3</td>
<td></td>
</tr>
<tr>
<td>S6:1,485</td>
<td></td>
</tr>
<tr>
<td>S6:1,486</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>S6:2,1</td>
<td></td>
</tr>
<tr>
<td>S6:2,2</td>
<td></td>
</tr>
<tr>
<td>S6:2,3</td>
<td></td>
</tr>
<tr>
<td>S6:2,485</td>
<td></td>
</tr>
<tr>
<td>S6:2,486</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1:1,1</td>
<td>S5:1,1</td>
</tr>
<tr>
<td>S1:1,2</td>
<td>S5:1,2</td>
</tr>
<tr>
<td>S1:1,3</td>
<td>S5:1,3</td>
</tr>
<tr>
<td>S1:1,243</td>
<td>S5:1,485</td>
</tr>
<tr>
<td>S1:1,486</td>
<td>S5:1,486</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1:2,1</td>
<td>S5:2,1</td>
</tr>
<tr>
<td>S1:2,2</td>
<td>S5:2,2</td>
</tr>
<tr>
<td>S1:2,3</td>
<td>S5:2,3</td>
</tr>
<tr>
<td>S1:2,243</td>
<td>S5:2,485</td>
</tr>
<tr>
<td>S1:2,486</td>
<td>S5:2,486</td>
</tr>
</tbody>
</table>

**KNOWN PROBLEMS OR ISSUES:**

None

**Dimension definitions:**

...
nscan1 var Number of scans in Swath S1 in the granule.
nscan2 var Number of scans in Swath S2 in the granule.
nscan3 var Number of scans in Swath S3 in the granule.
nscan4 var Number of scans in Swath S4 in the granule.
nscan5 var Number of scans in Swath S5 in the granule.
nscan6 var Number of scans in Swath S6 in the granule.
npixel1 243 Number of Swath S1 pixels in one scan.
npixel2 243 Number of Swath S2 pixels in one scan.
npixel3 243 Number of Swath S3 pixels in one scan.
npixel4 243 Number of Swath S4 pixels in one scan.
npixel5 486 Number of Swath S5 pixels in one scan.
npixel6 486 Number of Swath S6 pixels in one scan.
nchannel1 2 Number of Swath S1 channels.
nchannel2 2 Number of Swath S2 channels.
nchannel3 2 Number of Swath S3 channels.
nchannel4 2 Number of Swath S4 channels.
nchannel5 2 Number of Swath S5 channels.
nchannel6 2 Number of Swath S6 channels.
nchUIA1 1 Number of Swath S1 unique incidence angles.
nchUIA2 1 Number of Swath S2 unique incidence angles.
nchUIA3 1 Number of Swath S3 unique incidence angles.
nchUIA4 1 Number of Swath S4 unique incidence angles.
nchUIA5 1 Number of Swath S5 unique incidence angles.
nchUIA6 1 Number of Swath S6 unique incidence angles.

Figure 370 through Figure 388 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 370: Data Format Structure for 1CAMSR2, Common Calibrated Brightness Temperature

Figure 371: Data Format Structure for 1CAMSR2, S1
Figure 372: Data Format Structure for 1CAMS R2, S2

Figure 373: Data Format Structure for 1CAMS R2, S3
5.26 1CAMSR2 - Common Calibrated Brightness Temperature

Figure 374: Data Format Structure for 1CAMSR2, S4

Figure 375: Data Format Structure for 1CAMSR2, S5
Figure 376: Data Format Structure for 1CAMSAR2, S6

Figure 377: Data Format Structure for 1CAMSAR2, S1, ScanTime

Figure 378: Data Format Structure for 1CAMSAR2, S1, SCstatus
Figure 379: Data Format Structure for 1CAMSR2, S2, ScanTime

Figure 380: Data Format Structure for 1CAMSR2, S2, SCstatus

Figure 381: Data Format Structure for 1CAMSR2, S3, ScanTime
Figure 382: Data Format Structure for ICAMS/R2, S3, SCstatus

Figure 383: Data Format Structure for ICAMS/R2, S4, ScanTime

Figure 384: Data Format Structure for ICAMS/R2, S4, SCstatus
Figure 385: Data Format Structure for 1CAMSR2, S5, ScanTime
FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.
**NavigationRecord** (Metadata):  
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):  
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**XCALinfo** (Metadata):  
XCALinfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

**S1** (Swath)

**S1_SwathHeader** (Metadata):  
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S1_IncidenceAngleIndex** (Metadata):  
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S1)  
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan1):  
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan1):  
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan1):  
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan1):  
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan1):  
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value
Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

Longitude (4-byte float, array size: npixel1 x nscan1):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

SCstatus (Group in S1)

SCorientation (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel1 x nsan1):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-(1-98) = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

incidenceAngle (4-byte float, array size: nchUIA1 x npixel1 x nsan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUIA1 x npixel1 x nsan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.
incidenceAngleIndex (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

\[
\begin{align*}
\text{incidenceAngle}(2, npixel, nscan) \\
\text{sunGlintAngle}(2, npixel, nscan) \\
\text{incidenceAngleIndex}(10, nscan) \\
\text{Tc}(10, npixel, nscan)
\end{align*}
\]

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
\begin{align*}
i &= \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
\text{ia} &= \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
\text{sga} &= \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\end{align*}
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

Tc (4-byte float, array size: nchannel1 x npixel1 x nsan1):
GPM Common Calibrated Brightness Temperature. The channels are:

10.65 GHz vertically-polarized TBs
10.65 GHz horizontally-polarized TBs

S2 (Swath)
S2_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2
data products. See Metadata for GPM Products for details.

S2_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint
angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S2)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined
as:
-9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value
Latitude (4-byte float, array size: npixel1 x nscan1):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

Longitude (4-byte float, array size: npixel1 x nscan1):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

SCstatus (Group in S2)

SCorientation (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
- -9999 Missing value

SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
- -9999.9 Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
- -9999.9 Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
- -9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
- -9999.9 Missing value

Quality (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-(1-98) = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

incidenceAngle (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel1 x nscan1):
Index (1 based as in Fortran) of the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
i = \text{incidenceAngleIndex}(\text{channel}, \text{scan})
\]
\[
ia = \text{incidenceAngle}(i, \text{pixel}, \text{scan})
\]
\[
sga = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

**Tc** (4-byte float, array size: nchannel2 x npixel1 x nscan1):
GPM Common Calibrated Brightness Temperature. The channels are:

- 18.7 GHz vertically-polarized TBs
- 18.7 GHz horizontally-polarized TBs

**S3** (Swath)

**S3_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S3_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S3)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value
Month (1-byte integer, array size: nsan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nsan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nsan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nsan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nsan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nsan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nsan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nsan1):
A time associated with the scan. scanTime sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: npixel1 x nsan1):
Nominal latitude of the observation point on the earth surface at low frequency. This was
calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

Longitude (4-byte float, array size: npixel1 x nsan1):
Nominal longitude of the observation point on the earth surface at low frequency. This was
calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

SCstatus (Group in S3)

SCorientation (2-byte integer, array size: nsan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. The relationship of v to the sensor geometry is defined
in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values
are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
  1-99 = Generic flags (all sensors)
  100-127 = Sensor specific flags
1t 0 = Major errors resulting in missing data
  -(1-98) = Generic flags (all sensors)
  -99 = Missing value (no quality information available)
  -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

incidenceAngle (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

    incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

    i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.
Values range from 0 to 100. Special values are defined as:
-99  Missing value

$T_c$ (4-byte float, array size: nchannel3 x npixel1 x nscan1):
GPM Common Calibrated Brightness Temperature. The channels are:

23.8  GHz vertically-polarized TBs
23.8  GHz horizontally-polarized TBs

S4 (Swath)

S4_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S4_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indexes of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S4)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value
**Second** (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

**MilliSecond** (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

**DayOfYear** (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

**SecondOfDay** (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

**Latitude** (4-byte float, array size: npixel1 x nscan1):
Nominal latitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

**Longitude** (4-byte float, array size: npixel1 x nscan1):
Nominal longitude of the observation point on the earth surface at low frequency. This was calculated by applying the 23 GHz coregistration parameters to the 89A GHz location.

**SCstatus** (Group in S4)

**SCorientation** (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999  Missing value

**SClatitude** (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**SClongitude** (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**SCaltitude** (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9  Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**Quality** (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

---

**GENERAL SPECIFICATIONS:**
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-(1-98) = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

---

**DETAILED SPECIFICATIONS:**
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

---

**incidenceAngle** (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel
location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to
127 degrees. Sun below horizon value is -88. Missing value is -99.
incidenceAngleIndex (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
i = \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
i_a = \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
sga = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

Tc (4-byte float, array size: nchannel4 x npixel1 x nscan1): GPM Common Calibrated Brightness Temperature. The channels are:

36.5 GHz vertically-polarized TBs
36.5 GHz horizontally-polarized TBs

S5 (Swath)
S5_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S5_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S5)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan5):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- -9999 Missing value

Month (1-byte integer, array size: nscan5):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan5):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan5):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan5):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan5):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan5):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan5):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan5):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value
**Latitude** (4-byte float, array size: npixel5 x nscan5):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixel5 x nscan5):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCstatus** (Group in S5)

**SCorientation** (2-byte integer, array size: nscan5):
The angle of the spacecraft vector \(v\) from the satellite forward direction of motion, measured clockwise facing down. The relationship of \(v\) to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

**SClatitude** (4-byte float, array size: nscan5):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**SClongitude** (4-byte float, array size: nscan5):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCaltitude** (4-byte float, array size: nscan5):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan5):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**Quality** (1-byte integer, array size: npixel5 x nscan5):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**

0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
   1-99 = Generic flags (all sensors)
   100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
   -(1-98) = Generic flags (all sensors)
   -99 = Missing value (no quality information available)
   -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km
       threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

incidenceAngle (4-byte float, array size: nchUIA5 x npixel5 x nscan5):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
   -9999.9  Missing value

sunGlintAngle (1-byte integer, array size: nchUIA5 x npixel5 x nscan5):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel5 x nscan5):

   Index (1 based as in Fortran) of the incidence angle array corresponding to the channel.
   For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

   incidenceAngle(2,npixel,nscan)
sunGlintAngle(2, npixel, nscan)
incidenceAngleIndex(10, nscan)
Tc(10, npixel, nscan)

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```python
i = incidenceAngleIndex(channel, scan)
ia = incidenceAngle(i, pixel, scan)
sga = sunGlintAngle(i, pixel, scan)
```

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99   Missing value

Tc (4-byte float, array size: nchannel x npixel x nscan):
GPM Common Calibrated Brightness Temperature. The channels are:

89 GHz vertically-polarized TBs
89 GHz horizontally-polarized TBs

S6 (Swath)

S6_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S6_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indices of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S6)
A UTC time associated with the scan.
Year (2-byte integer, array size: nscan6):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
   -9999 Missing value

Month (1-byte integer, array size: nscan6):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
   -99 Missing value

DayOfMonth (1-byte integer, array size: nscan6):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
   -99 Missing value

Hour (1-byte integer, array size: nscan6):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
   -99 Missing value

Minute (1-byte integer, array size: nscan6):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
   -99 Missing value

Second (1-byte integer, array size: nscan6):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
   -99 Missing value

MilliSecond (2-byte integer, array size: nscan6):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
   -9999 Missing value

DayOfYear (2-byte integer, array size: nscan6):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
   -9999 Missing value

SecondOfDay (8-byte float, array size: nscan6):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
   -9999.9 Missing value

Latitude (4-byte float, array size: npixel6 x nscan6):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
   -9999.9 Missing value

Longitude (4-byte float, array size: npixel6 x nscan6):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9 Missing value
SCstatus (Group in S6)

SCorientation (2-byte integer, array size: nscan6):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan6):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan6):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan6):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan6):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel6 x nscan6):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
- 0 = Good data in all channels in the swath
- gt 0 = Cautionary warning flags
  - 1-99 = Generic flags (all sensors)
  - 100-127 = Sensor specific flags
- lt 0 = Major errors resulting in missing data
  - -(1-98) = Generic flags (all sensors)
  - -99 = Missing value (no quality information available)
  - -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
- 1 = Possible sunGlint, 0 le sunGlintAngle lt 20
- 2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

**incidenceAngle** (4-byte float, array size: nchUIA6 x npixel6 x nscan6):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA6 x npixel6 x nscan6):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel6 x nscan6):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

```
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
```

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```
i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)
```
The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99   Missing value

**\( T_c \)** (4-byte float, array size: nchannel6 x npixel6 x nscan6):
GPM Common Calibrated Brightness Temperature. The channels are:

- 89 GHz vertically-polarized TBs
- 89 GHz horizontally-polarized TBs

**C Structure Header file:**

```c
#ifndef _TK_1CAMSR2_H_
#define _TK_1CAMSR2_H_

#ifndef _L1CAMSR2_S6_
#define _L1CAMSR2_S6_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[486];
    float Longitude[486];
    SCSTATUS SCstatus;
    signed char Quality[486];
    float incidenceAngle[486][1];
    signed char sunGlintAngle[486][1];
    signed char incidenceAngleIndex[2];
    float Tc[486][2];
} L1CAMSR2_S6;
#endif

#endif

#ifndef _L1CAMSR2_S5_
#define _L1CAMSR2_S5_

typedef struct {
```
```
typedef struct {
    SCANTIME ScanTime;
    float Latitude[243];
    float Longitude[243];
    SCSTATUS SCstatus;
    signed char Quality[243];
    float incidenceAngle[243][1];
    signed char sunGlintAngle[243][1];
    signed char incidenceAngleIndex[2];
    float Tc[243][2];
} L1CAMSR2_S4;

#endif
#ifndef _L1CAMSR2_S3_
#define _L1CAMSR2_S3_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[243];
    float Longitude[243];
    SCSTATUS SCstatus;
    signed char Quality[243];
    float incidenceAngle[243][1];
    signed char sunGlintAngle[243][1];
    signed char incidenceAngleIndex[2];
    float Tc[243][2];
} L1CAMSR2_S3;
#endif
#ifndef _L1CAMSR2_S2_
define _L1CAMSR2_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[243];
    float Longitude[243];
    SCSTATUS SCstatus;
    signed char Quality[243];
    float incidenceAngle[243][1];
    signed char sunGlintAngle[243][1];
    signed char incidenceAngleIndex[2];
    float Tc[243][2];
} L1CAMSR2_S2;
define _L1CAMSR2_S2_

#endif

#ifndef _SCSTATUS_
define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
define _SCSTATUS_

#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
} SCANTIME;
define _SCANTIME_

#endif
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L1CAMSR2_S1_
#define _L1CAMSR2_S1_

typedef struct {
  SCANTIME ScanTime;
  float Latitude[243];
  float Longitude[243];
  SCSTATUS SCstatus;
  signed char Quality[243];
  float incidenceAngle[243][1];
  signed char sunGlintAngle[243][1];
  signed char incidenceAngleIndex[2];
  float Tc[243][2];
} L1CAMSR2_S1;
#endif

#ifndef _L1CAMSR2_SWATHS_
#define _L1CAMSR2_SWATHS_

typedef struct {
  L1CAMSR2_S1 S1;
  L1CAMSR2_S2 S2;
  L1CAMSR2_S3 S3;
  L1CAMSR2_S4 S4;
  L1CAMSR2_S5 S5;
  L1CAMSR2_S6 S6;
} L1CAMSR2_SWATHS;
#endif

#endif

Fortran Structure Header file:
STRUCTURE /L1CAMSR2_S6/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(486)
    REAL*4 Longitude(486)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(486)
    REAL*4 incidenceAngle(1,486)
    BYTE sunGlintAngle(1,486)
    BYTE incidenceAngleIndex(2)
    REAL*4 Tc(2,486)
END STRUCTURE

STRUCTURE /L1CAMSR2_S5/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(486)
    REAL*4 Longitude(486)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(486)
    REAL*4 incidenceAngle(1,486)
    BYTE sunGlintAngle(1,486)
    BYTE incidenceAngleIndex(2)
    REAL*4 Tc(2,486)
END STRUCTURE

STRUCTURE /L1CAMSR2_S4/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(243)
    REAL*4 Longitude(243)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(243)
    REAL*4 incidenceAngle(1,243)
    BYTE sunGlintAngle(1,243)
    BYTE incidenceAngleIndex(2)
    REAL*4 Tc(2,243)
END STRUCTURE

STRUCTURE /L1CAMSR2_S3/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(243)
    REAL*4 Longitude(243)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(243)
    REAL*4 incidenceAngle(1,243)
BYTE sunGlintAngle(1,243)
BYTE incidenceAngleIndex(2)
REAL*4 Tc(2,243)
END STRUCTURE

STRUCTURE /L1CAMSR2_S2/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(243)
  REAL*4 Longitude(243)
  RECORD /SCSTATUS/ SCstatus
  BYTE Quality(243)
  REAL*4 incidenceAngle(1,243)
  BYTE sunGlintAngle(1,243)
  BYTE incidenceAngleIndex(2)
  REAL*4 Tc(2,243)
END STRUCTURE

STRUCTURE /SCSTATUS/
  INTEGER*2 SCorientation
  REAL*4 SClatitude
  REAL*4 SClongitude
  REAL*4 SCaltitude
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1CAMSR2_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(243)
  REAL*4 Longitude(243)
  RECORD /SCSTATUS/ SCstatus
  BYTE Quality(243)
5.27 1CWIND - Common Calibrated Brightness Temperature

1CWIND contains common calibrated brightness temperature from the WindSat passive microwave instrument flown on the Coriolis satellite. Swath S1 is the only swath and has 8 channels (10.7V 10.7H 18.7V 18.7H 23.8V 23.8H 37V 37H) All the above frequencies are in GHz.

This data is a subset of the WindSat Sensor Data Record (SDR) files. Only the V and H polarizations in the direction of forward motion are included in the Level 1C files, resulting in a total of 8 channels with 80 pixels per scan. The source data have been remapped to the spacing of every fourth 37 GHz VH observation along scan and every scan along track resulting in approximately a 12.5 x 12.5 km grid locally. The swath of the source data is defined as the common swath of all channels at 10.7, 18.7, 23.8, and 37.0 GHz in the forward direction. This swath is a 68° segment of the observations in the forward direction. We define the spacecraft vector (v) at the center of this forward observation segment. "v" is used in the definition of the variable SCorientation.

RELATION BETWEEN THE SWATHS: S1 is the only swath, containing observations sampled 80 times along the scan.

KNOWN PROBLEMS OR ISSUES:
1. The ScanTime can be late by as much as 0.35 s in the event of missing observations. The source data does not report time for each scan, but reports time for each SDR along the scan. We chose to use time of the first non-missing SDR in a scan. If some SDRs are missing in the beginning of the scan, the ScanTime will be late. We could have instead calculated the time of the first observation in the scan, had it not been missing. However, in that case the spacecraft latitude and longitude (taken from the first SDR in the scan) would be incorrect for the calculated ScanTime.

Dimension definitions:
Figure 389: Data Format Structure for 1CWIND, Common Calibrated Brightness Temperature

Figure 390: Data Format Structure for 1CWIND, S1

nscan1 var Number of Swath 1 scans in the granule.
nchannel1 8 Number of Swath 1 channels.
npixel1 80 Number of Swath 1 pixels in one scan.
nchUIA1 4 Number of Swath S1 unique incidence angles.

Figure 389 through Figure 392 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information
Figure 391: Data Format Structure for 1CWIND, ScanTime

Figure 392: Data Format Structure for 1CWIND, SCstatus
separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**XCALinfo** (Metadata):
XCALinfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

---

**S1** (Swath)

**S1_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S1_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

---

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
**Minute** (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

**Second** (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

**MilliSecond** (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

**DayOfYear** (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

**SecondOfDay** (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

**Latitude** (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**Longitude** (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**SCstatus** (Group)

**SCorientation** (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999  Missing value

**SClatitude** (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value
SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
- 0 = Good data in all channels in the swath
- gt 0 = Cautionary warning flags
  - 1-99 = Generic flags (all sensors)
  - 100-127 = Sensor specific flags
- lt 0 = Major errors resulting in missing data
  - -(1-98) = Generic flags (all sensors)
  - -99 = Missing value (no quality information available)
  - -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
- 1 = Possible sunGlint, 0 le sunGlintAngle lt 20
- 2 = Possible radio frequency interference
- 3 = Degraded geolocation data
- 4 = Data corrected for warm load instrusion

100 = solar disturbance of warm load
101 = corrected warm load gains applied
102 = cold load flag, RFI or moon contamination present in scan for 10.7GHz
103 = cold load flag, RFI or moon contamination present in scan for 18.7GHz
104 = cold load flag, RFI or moon contamination present in scan for 23.8GHz
105 = cold load flag, RFI or moon contamination present in scan for 37.0GHz
106 = warm load flag, thermal gradients present for 10.7GHz
107 = warm load flag, thermal gradients present for 18.7GHz
108 = warm load flag, thermal gradients present for 23.8GHz
109 = warm load flag, thermal gradients present for 37.0GHz
110 = resampling percentage threshold was not met for 10V
111 = resampling percentage threshold was not met for 10H
112 = resampling percentage threshold was not met for 18V
113 = resampling percentage threshold was not met for 18H
114 = resampling percentage threshold was not met for 23V
115 = resampling percentage threshold was not met for 23H
116 = resampling percentage threshold was not met for 37V
117 = resampling percentage threshold was not met for 37H

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb < 50 or Tb > 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

-100 = gain saturation, strong RFI causes gain to change for 10.7GHz
-101 = gain saturation, strong RFI causes gain to change for 18.7GHz
-102 = gain saturation, strong RFI causes gain to change for 23.8GHz
-103 = gain saturation, strong RFI causes gain to change for 37.0GHz
-104 = satellite attitude transient
-105 = star viewer outage near attitude transient

incidenceAngle (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions
in Fortran would be:

\[
\begin{align*}
\text{incidenceAngle}(2, \text{npixel}, \text{nscan}) \\
\text{sunGlintAngle}(2, \text{npixel}, \text{nscan}) \\
\text{incidenceAngleIndex}(10, \text{nscan}) \\
\text{Tc}(10, \text{npixel}, \text{nscan})
\end{align*}
\]

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
\begin{align*}
i & = \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
\text{ia} & = \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
\text{sga} & = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\end{align*}
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:

-99 Missing value

\text{Tc} (4-byte float, array size: \text{nchannel1 x npixel1 x nscan1}):
GPM Common Calibrated Brightness Temperature. The channels are:

10.7 GHz vertically-polarized TBs
10.7 GHz horizontally-polarized TBs
18.7 GHz vertically-polarized TBs
18.7 GHz horizontally-polarized TBs
23.8 GHz vertically-polarized TBs
23.8 GHz horizontally-polarized TBs
37.0 GHz vertically-polarized TBs
37.0 GHz horizontally-polarized TBs

C Structure Header file:

```c
#ifndef _TK_1CWIND_H_
#define _TK_1CWIND_H_

#ifndef _SCSTATUS_
```


#define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;

#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L1CWIND_S1_
#define _L1CWIND_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[80];
    float Longitude[80];
    SCSTATUS SCstatus;
    signed char Quality[80];
    float incidenceAngle[80][4];
    signed char sunGlintAngle[80][4];
    signed char incidenceAngleIndex[8];
    float Tc[80][8];
} L1CWIND_S1;
#ifndef _L1CWIND_SWATHS_
define _L1CWIND_SWATHS_

typedef struct {  
    L1CWIND_S1 S1;  
} L1CWIND_SWATHS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation  
    REAL*4 SClatitude  
    REAL*4 SClongitude  
    REAL*4 SCaltitude  
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year  
    BYTE Month  
    BYTE DayOfMonth  
    BYTE Hour  
    BYTE Minute  
    BYTE Second  
    INTEGER*2 MilliSecond  
    INTEGER*2 DayOfYear  
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1CWIND_S1/
    RECORD /SCANTIME/ ScanTime  
    REAL*4 Latitude(80)  
    REAL*4 Longitude(80)  
    RECORD /SCSTATUS/ SCstatus  
    BYTE Quality(80)  
    REAL*4 incidenceAngle(4,80)
5.28 1CMHS - Common Calibrated Brightness Temperature

1CMHS contains common calibrated brightness temperature from the MHS passive microwave instrument flown on the NOAA and METOPS satellites. Swath S1 is the only swath and has 5 channels (89.0GHzV, 157.0GHzV, 183.31GHz+/-1GHzH, 183.31GHz+/-3GHzH, and 190.31GHzV). MHS is very similar to AMSU-B. The scan period is 2.667s.

RELATION BETWEEN THE SWATHS: S1 is the only swath, containing observations sampled 90 times along the scan.

KNOWN PROBLEMS OR ISSUES WITH REVISION 1 DATA: None.

Dimension definitions:
- nscan1 var Number of Swath 1 scans in the granule.
- nchannel1 5 Number of Swath 1 channels.
- npixel1 90 Number of Swath 1 pixels in one scan.
- nchUIA1 1 Number of Swath S1 unique incidence angles.

Figure 393 through Figure 396 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See
**Figure 394: Data Format Structure for 1CMHS, S1**

**ScanTime**
- **Year**: 2 bytes (Array: nscan1)
- **Month**: 1 byte (Array: nscan1)
- **DayOfMonth**: 1 byte (Array: nscan1)
- **Hour**: 1 byte (Array: nscan1)
- **Minute**: 1 byte (Array: nscan1)
- **Second**: 1 byte (Array: nscan1)
- **MilliSecond**: 2 bytes (Array: nscan1)
- **DayOfYear**: 2 bytes (Array: nscan1)
- **SecondOfDay**: 8 bytes (Array: nscan1)

**SCstatus**
- **SCorientation**: 2 bytes (Array: nscan1)
- **SClatitude**: 4 bytes (Array: nscan1)
- **SClongitude**: 4 bytes (Array: nscan1)
- **SCaltitude**: 4 bytes (Array: nscan1)
- **FractionalGranuleNumber**: 8 bytes (Array: nscan1)

**Figure 395: Data Format Structure for 1CMHS, ScanTime**

**Figure 396: Data Format Structure for 1CMHS, SCstatus**
Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**XCALInfo (Metadata):**
XCALInfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

**S1 (Swath)**

**S1_SwathHeader (Metadata):**
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S1_IncidenceAngleIndex (Metadata):**
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime (Group)**
A UTC time associated with the scan.

**Year (2-byte integer, array size: nscan1):**
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month (1-byte integer, array size: nscan1):**
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth (1-byte integer, array size: nscan1):**
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value
Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group)

SCorientation (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value
SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
  0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
  1-99 = Generic flags (all sensors)
  100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
  -(1-98) = Generic flags (all sensors)
  -99 = Missing value (no quality information available)
  -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
  1 = Possible sunGlint, 0 le sunGlintAngle lt 20
  2 = Possible radio frequency interference
  3 = Degraded geolocation data
  4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

**incidenceAngle** (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel1 x nscan1):
Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

```plaintext
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
```

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```plaintext
i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)
```

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value
**Tc (4-byte float, array size: nchannel1 x npixel1 x nscan1):**
GPM Common Calibrated Brightness Temperature. The channels are:

- 89.0 GHz vertically-polarized TBs
- 157.0 GHz vertically-polarized TBs
- 183.31 GHz +/-1GHz horizontally-polarized TBs
- 183.31 GHz +/-3GHz horizontally-polarized TBs
- 190.31 GHz vertically-polarized TBs

**C Structure Header file:**

```c
#ifndef _TK_1CMHS_H_
#define _TK_1CMHS_H_

#ifndef _SCSTATUS_
#define _SCSTATUS_

typedef struct {
  short SCorientation;
  float SClatitude;
  float SClongitude;
  float SCaltitude;
  double FractionalGranuleNumber;
} SCSTATUS;

#endif
#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
  short Year;
  signed char Month;
  signed char DayOfMonth;
  signed char Hour;
  signed char Minute;
  signed char Second;
  short MilliSecond;
  short DayOfYear;
  double SecondOfDay;
} SCANTIME;

#endif
```

```c
#ifndef _L1CMHS_S1_
define _L1CMHS_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[90];
    float Longitude[90];
    SCSTATUS SCstatus;
    signed char Quality[90];
    float incidenceAngle[90][1];
    signed char sunGlintAngle[90][1];
    signed char incidenceAngleIndex[5];
    float Tc[90][5];
} L1CMHS_S1;
#endif

#ifndef _L1CMHS_SWATHS_
define _L1CMHS_SWATHS_

typedef struct {
    L1CMHS_S1 S1;
} L1CMHS_SWATHS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude
    REAL*4 SCaltitude
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
```
5.29 1CSAPHIR - Common Calibrated Brightness Temperature

1CSAPHIR contains common calibrated brightness temperature from the SAPHIR passive microwave instrument flown on the Megha-Tropiques satellite. Swath S1 is the only swath and has 6 channels (S1 S2 S3 S4 S5 S6) The channels are 183.31 +/- delta GHz, where delta = 0.2, 1.1, 2.8, 4.2, 6.8, 11.0.

RELATION BETWEEN THE SWATHS: S1 is the only swath, containing observations sampled 182 times along the scan.

KNOWN PROBLEMS OR ISSUES WITH REVISION 1 DATA: None.

Dimension definitions:

nscan1 var Number of Swath 1 scans in the granule.
nchannel1 6 Number of Swath 1 channels.
npixel1 182 Number of Swath 1 pixels in one scan.
nchUIA1 1 Number of Swath S1 unique incidence angles.

Figure 397 through Figure 400 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 397: Data Format Structure for 1CSAPHIR, Common Calibrated Brightness Temperature

Figure 398: Data Format Structure for 1CSAPHIR, S1

Figure 399: Data Format Structure for 1CSAPHIR, ScanTime
Figure 400: Data Format Structure for 1CSAPHIR, SCstatus

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**XCALInfo** (Metadata):
XCALInfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

### S1 (Swath)

**S1_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S1_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group)
A UTC time associated with the scan.
Year (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
SCstatus (Group)

SCorientation (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-(1-98) = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

101 = Backward scanning

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

-100 = Invalid scan
-101 = Scan error
-102 = date/time error
-103 = PRT error
-104 = CRC error
-105 = Payload not nominal
-110 = Channel is off
-111 = L0 count saturated or has poor value
-112 = Hot/cold count not available
-113 = Calibration issue

incidenceAngle (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

Tc (4-byte float, array size: nchannel1 x npixel1 x nscan1):
GPM Common Calibrated Brightness Temperature. The channels are:

(S1 S2 S3 S4 S5 S6)
183.31 +/- delta GHz, where
delta = 0.2, 1.1, 2.8, 4.2, 6.8, 11.0.

C Structure Header file:

```c
#ifndef _TK_1CSAPHIR_H_
define _TK_1CSAPHIR_H_
#endif

#ifndef _SCSTATUS_
define _SCSTATUS_
#endif
typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
```
float SCAltitude;
double FractionalGranuleNumber;
} SCSTATUS;
#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
  short Year;
signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L1CSAPHIR_S1_
#define _L1CSAPHIR_S1_

typedef struct {
  SCANTIME ScanTime;
  float Latitude[182];
  float Longitude[182];
  SCSTATUS SCstatus;
signed char Quality[182];
  float incidenceAngle[182][1];
signed char sunGlintAngle[182][1];
signed char incidenceAngleIndex[6];
  float Tc[182][6];
} L1CSAPHIR_S1;
#endif

#ifndef _L1CSAPHIR_SWATHS_
#define _L1CSAPHIR_SWATHS_


typedef struct {
    L1CSAPHIR_S1 S1;
} L1CSAPHIR_SWATHS;

#endif
#endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude
    REAL*4 SCaltitude
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    BYTE MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1CSAPHIR_S1/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(182)
    REAL*4 Longitude(182)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(182)
    REAL*4 incidenceAngle(1,182)
    BYTE sunGlintAngle(1,182)
    BYTE incidenceAngleIndex(6)
    REAL*4 Tc(6,182)
END STRUCTURE

STRUCTURE /L1CSAPHIR_SWATHS/
5.30 1CATMS - Common Calibrated Brightness Temperature

1CATMS contains common calibrated brightness temperature from the ATMS passive microwave instrument flown on the Suomi NPP satellite and JPSS satellites. ATMS is approximately a combination of the AMSU-A channels and the MHS channels. ATMS rotates 3 scans per 8 seconds. ATMS has the following 22 channels:

<table>
<thead>
<tr>
<th>Ch</th>
<th>GHz</th>
<th>Pol</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>23.8</td>
<td>QV</td>
</tr>
<tr>
<td>2</td>
<td>31.4</td>
<td>QV</td>
</tr>
<tr>
<td>3</td>
<td>50.3</td>
<td>QH</td>
</tr>
<tr>
<td>4</td>
<td>51.76</td>
<td>QH</td>
</tr>
<tr>
<td>5</td>
<td>52.8</td>
<td>QH</td>
</tr>
<tr>
<td>6</td>
<td>53.596+-0.115</td>
<td>QH</td>
</tr>
<tr>
<td>7</td>
<td>54.4</td>
<td>QH</td>
</tr>
<tr>
<td>8</td>
<td>54.94</td>
<td>QH</td>
</tr>
<tr>
<td>9</td>
<td>55.5</td>
<td>QH</td>
</tr>
<tr>
<td>10</td>
<td>fo = 57.29</td>
<td>QH</td>
</tr>
<tr>
<td>11</td>
<td>fo+-0.3222+-0.217</td>
<td>QH</td>
</tr>
<tr>
<td>12</td>
<td>fo+-0.3222+-0.048</td>
<td>QH</td>
</tr>
<tr>
<td>13</td>
<td>fo+-0.3222+-0.022</td>
<td>QH</td>
</tr>
<tr>
<td>14</td>
<td>fo+-0.3222+-0.010</td>
<td>QH</td>
</tr>
<tr>
<td>15</td>
<td>fo+-0.3222+-0.0045</td>
<td>QH</td>
</tr>
<tr>
<td>16</td>
<td>88.2</td>
<td>QV</td>
</tr>
<tr>
<td>17</td>
<td>165.5</td>
<td>QH</td>
</tr>
<tr>
<td>18</td>
<td>183.31+-7</td>
<td>QH</td>
</tr>
<tr>
<td>19</td>
<td>183.31+-4.5</td>
<td>QH</td>
</tr>
<tr>
<td>20</td>
<td>183.31+-3</td>
<td>QH</td>
</tr>
<tr>
<td>21</td>
<td>183.31+-1.8</td>
<td>QH</td>
</tr>
<tr>
<td>22</td>
<td>183.31+-1</td>
<td>QH</td>
</tr>
</tbody>
</table>

QV means quasi-vertical; the polarization vector is parallel to the scan plane at nadir.
QH means quasi-horizontal polarization.

Note on geolocation and 1C swaths:
The BeamLatitude and BeamLongitude in 1BASEATMS
have a band dimension of 5. Lat and lon is for channels 1,2,3,16,17. Each 1C swath will contain one band:

<table>
<thead>
<tr>
<th>1C swath</th>
<th>Band</th>
<th>IEEE GHz</th>
<th>Ch geo</th>
<th>Chs in band</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>K</td>
<td>18-26.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>A(Ka)</td>
<td>26.5-40</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>W</td>
<td>75-110</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>4</td>
<td>G</td>
<td>110-300</td>
<td>17</td>
<td>17-22</td>
</tr>
</tbody>
</table>

Note that channels 3-15 are NOT included in the 1C product.

1CATMS contains 4 swaths, one for each band K, A(Ka), W, and G.

RELATION BETWEEN THE SWATHS: All 4 swaths contain observations sampled 96 times along the scan.

KNOWN PROBLEMS OR ISSUES WITH REVISION 1 DATA: None.

Dimension definitions:

- `nscan1` var Number of Swath 1 scans in the granule.
- `nchannel1` 1 Number of Swath 1 channels.
- `npixel1` 96 Number of Swath 1 pixels in one scan.
- `nchUIA1` 1 Number of Swath S1 unique incidence angles.
- `nscan2` var Number of Swath 2 scans in the granule.
- `nchannel2` 1 Number of Swath 2 channels.
- `npixel2` 96 Number of Swath 2 pixels in one scan.
- `nchUIA2` 1 Number of Swath S2 unique incidence angles.
- `nscan3` var Number of Swath 3 scans in the granule.
- `nchannel3` 1 Number of Swath 3 channels.
- `npixel3` 96 Number of Swath 3 pixels in one scan.
- `nchUIA3` 1 Number of Swath S3 unique incidence angles.
- `nscan4` var Number of Swath 4 scans in the granule.
- `nchannel4` 6 Number of Swath 4 channels.
- `npixel4` 96 Number of Swath 4 pixels in one scan.
- `nchUIA4` 1 Number of Swath S4 unique incidence angles.

Figure 401 through Figure 413 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.
Figure 401: Data Format Structure for 1CATMS, Common Calibrated Brightness Temperature

Figure 402: Data Format Structure for 1CATMS, S1
Figure 403: Data Format Structure for 1CATMS, S2

Figure 404: Data Format Structure for 1CATMS, S3
5 STANDARD GPM PRODUCTS

Figure 405: Data Format Structure for 1CATMS, S4

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Array Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>S4_SwathHeader</td>
<td>Metadata</td>
<td></td>
</tr>
<tr>
<td>S4_IncidenceAngleIndex</td>
<td>Metadata</td>
<td></td>
</tr>
<tr>
<td>ScanTime</td>
<td>19 bytes</td>
<td>Group: nscan4</td>
</tr>
<tr>
<td>Latitude</td>
<td>4 bytes</td>
<td>Array: npixel4 x nscan4</td>
</tr>
<tr>
<td>Longitude</td>
<td>4 bytes</td>
<td>Array: npixel4 x nscan4</td>
</tr>
<tr>
<td>SCstatus</td>
<td>22 bytes</td>
<td>Array: nscan4</td>
</tr>
<tr>
<td>Quality</td>
<td>1 byte</td>
<td>Array: npixel4 x nscan4</td>
</tr>
<tr>
<td>IncidenceAngle</td>
<td>4 bytes</td>
<td>Array: nchUIA4 x npixel4 x nscan4</td>
</tr>
<tr>
<td>SunGlintAngle</td>
<td>1 byte</td>
<td>Array: nchUIA4 x npixel4 x nscan4</td>
</tr>
<tr>
<td>IncidenceAngleIndex</td>
<td>1 byte</td>
<td>Array: nchannel4 x nscan4</td>
</tr>
<tr>
<td>Tc</td>
<td>4 bytes</td>
<td>Array: nchannel4 x npixel4 x nscan4</td>
</tr>
</tbody>
</table>

Figure 406: Data Format Structure for 1CATMS, S1, ScanTime

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Array Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScanTime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Year</td>
<td>2 bytes</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>Month</td>
<td>1 byte</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>DayOfMonth</td>
<td>1 byte</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>Hour</td>
<td>1 byte</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>Minute</td>
<td>1 byte</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>Second</td>
<td>1 byte</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>MilliSecond</td>
<td>2 bytes</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>DayOfYear</td>
<td>2 bytes</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>SecondOfDay</td>
<td>8 bytes</td>
<td>Array: nscan1</td>
</tr>
</tbody>
</table>

Figure 407: Data Format Structure for 1CATMS, S1, SCstatus

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Array Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>SC_orientation</td>
<td>2 bytes</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>SC_latitude</td>
<td>4 bytes</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>SC_longitude</td>
<td>4 bytes</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>SC_altitude</td>
<td>4 bytes</td>
<td>Array: nscan1</td>
</tr>
<tr>
<td>FractionalGranuleNumber</td>
<td>8 bytes</td>
<td>Array: nscan1</td>
</tr>
</tbody>
</table>
5.30 1CATMS - Common Calibrated Brightness Temperature

![Diagram](image-url)

**Figure 408:** Data Format Structure for 1CATMS, S2, ScanTime

**Figure 409:** Data Format Structure for 1CATMS, S2, SCstatus

**Figure 410:** Data Format Structure for 1CATMS, S3, ScanTime
Figure 411: Data Format Structure for 1CATMS, S3, SCstatus

Figure 412: Data Format Structure for 1CATMS, S4, ScanTime

Figure 413: Data Format Structure for 1CATMS, S4, SCstatus
**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**XCALinfo** (Metadata):
XCALinfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

**S1** (Swath)

**S1_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S1_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S1)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan1):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan1):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan1):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value
**Hour** (1-byte integer, array size: nscan1):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan1):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCstatus** (Group in S1)

**SCorientation** (2-byte integer, array size: nscan1):
The angle of the spacecraft vector \( v \) from the satellite forward direction of motion, measured clockwise facing down. The relationship of \( v \) to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value
SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
1-99 = Generic flags (all sensors)
100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
-1-98 = Generic flags (all sensors)
-99 = Missing value (no quality information available)
-(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10  =  Distance to its corresponding LF pixel exceeds 7Km
threshold.  used in L1C-R product only
-99  =  Missing value (no quality information available)

-100 = Missing scan indicated by QF19\_SCAN\_ATMSSDR
-101 = Time sequence error
-102 = Insufficient KAV PRT data
-103 = Insufficient WG PRT data
-104 = Space view antenna position error
-105 = Blackbody view antenna position error

**incidenceAngle** (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel
location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to
127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of
the incidence angle array corresponding to the channel.
For example, if the swath has 10 channels and
2 unique incidence angles, then the dimensions
in Fortran would be:

\[
\text{incidenceAngle}(2, \text{npixel}, \text{nscan}) \\
\text{sunGlintAngle}(2, \text{npixel}, \text{nscan}) \\
\text{incidenceAngleIndex}(10, \text{nscan}) \\
\text{Tc}(10, \text{npixel}, \text{nscan})
\]

The user would do the following to retrieve the angles
for a given channel, pixel, and scan:

\[
i = \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
iA = \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
sGA = \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\]

The incidenceAngleIndex is the same for every scan,
but is repeated each scan for the convenience of users
reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

\( T_c \) (4-byte float, array size: nchannel1 x npixel1 x nscan1):
GPM Common Calibrated Brightness Temperature. The channels are:

\( 23.8 \ \text{GHz quasi vertically-polarized TBs} \)

**S2** (Swath)

**S2_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S2_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group in S2)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan2):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan2):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan2):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan2):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nscan2):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan2):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan2):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan2):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan2):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel2 x nscan2):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel2 x nscan2):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group in S2)

SCorientation (2-byte integer, array size: nscan2):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan2):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value
SClongitude (4-byte float, array size: nscan2):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan2):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan2):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel2 x nscan2):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
   1-99 = Generic flags (all sensors)
   100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
   -(1-98) = Generic flags (all sensors)
   -99 = Missing value (no quality information available)
   -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km
      threshold. used in L1C-R product only
-99 = Missing value (no quality information available)
incidenceAngle (4-byte float, array size: nchUIA2 x npixel2 x nscan2):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUIA2 x npixel2 x nscan2):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel2 x nscan2):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

\[
\begin{align*}
\text{incidenceAngle}(2, \text{npixel}, \text{nscan}) \\
\text{sunGlintAngle}(2, \text{npixel}, \text{nscan}) \\
\text{incidenceAngleIndex}(10, \text{nscan}) \\
\text{Tc}(10, \text{npixel}, \text{nscan})
\end{align*}
\]

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
\begin{align*}
i &= \text{incidenceAngleIndex}(\text{channel}, \text{scan}) \\
i_a &= \text{incidenceAngle}(i, \text{pixel}, \text{scan}) \\
s_ga &= \text{sunGlintAngle}(i, \text{pixel}, \text{scan})
\end{align*}
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99 Missing value

Tc (4-byte float, array size: nchannel2 x npixel2 x nscan2):
GPM Common Calibrated Brightness Temperature. The channels are:

\[
31.4 \text{ GHz quasi-vertically-polarized TBs}
\]
S3 (Swath)

S3_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

S3_IncidenceAngleIndex (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

ScanTime (Group in S3)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan3):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

Month (1-byte integer, array size: nscan3):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan3):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan3):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan3):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan3):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan3):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan3):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value
SecondOfDay (8-byte float, array size: nscan3):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel3 x nscan3):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel3 x nscan3):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group in S3)

SCorientation (2-byte integer, array size: nscan3):
The angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. The relationship of v to the sensor geometry is defined
in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values
are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan3):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan3):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan3):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan3):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel3 x nscan3):
Quality of Tc in the swath.
GENERAL SPECIFICATIONS:

0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
  1-99 = Generic flags (all sensors)
  100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
  -(1-98) = Generic flags (all sensors)
    -99 = Missing value (no quality information available)
  -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:

1 = Possible sunGlint, 0 ≤ sunGlintAngle < 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion
-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb ≤ 50 or Tb ≥ 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km
 threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

incidenceAngle (4-byte float, array size: nchannel x npixel x nscan):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel
location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
  -9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchannel x npixel x nscan):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to
127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel x nscan):

  Index (1 based as in Fortran) of
  the incidence angle array corresponding to the channel.
  For example, if the swath has 10 channels and
  2 unique incidence angles, then the dimensions
in Fortran would be:

```fortran
incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)
```

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

```fortran
i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i,pixel,scan)
sga = sunGlintAngle(i,pixel,scan)
```

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:

-99  Missing value

\( T_c \) (4-byte float, array size: nchannel3 x npixel3 x nscan3):
GPM Common Calibrated Brightness Temperature. The channels are:

- 88.2 GHz quasi-vertically-polarized TBs

**S4** (Swath)

**S4_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S4_IncidenceAngleIndex** (Metadata):
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.
ScanTime (Group in S4)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan4):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
   -9999  Missing value

Month (1-byte integer, array size: nscan4):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
   -99   Missing value

DayOfMonth (1-byte integer, array size: nscan4):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
   -99   Missing value

Hour (1-byte integer, array size: nscan4):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
   -99   Missing value

Minute (1-byte integer, array size: nscan4):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
   -99   Missing value

Second (1-byte integer, array size: nscan4):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
   -99  Missing value

MilliSecond (2-byte integer, array size: nscan4):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
   -9999   Missing value

DayOfYear (2-byte integer, array size: nscan4):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
   -9999   Missing value

SecondOfDay (8-byte float, array size: nscan4):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
   -9999.9   Missing value

Latitude (4-byte float, array size: npixel4 x nscan4):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
   -9999.9   Missing value

Longitude (4-byte float, array size: npixel4 x nscan4):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCstatus** (Group in S4)

**SCorientation** (2-byte integer, array size: nscan4):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

**SClatitude** (4-byte float, array size: nscan4):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**SClongitude** (4-byte float, array size: nscan4):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCaltitude** (4-byte float, array size: nscan4):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan4):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**Quality** (1-byte integer, array size: npixel4 x nscan4):
Quality of Tc in the swath.

**GENERAL SPECIFICATIONS:**

- 0 = Good data in all channels in the swath
- gt 0 = Cautionary warning flags
  - 1-99 = Generic flags (all sensors)
  - 100-127 = Sensor specific flags
- lt 0 = Major errors resulting in missing data
  - -(1-98) = Generic flags (all sensors)
  - -99 = Missing value (no quality information available)
  - -(100-127) = Sensor specific flags
DETAILED SPECIFICATIONS:

1 = Possible sunGlint, 0 ≤ sunGlintAngle < 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb < 50 or Tb > 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

incidenceAngle (4-byte float, array size: nchUIA4 x npixel4 x nscan4):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

sunGlintAngle (1-byte integer, array size: nchUIA4 x npixel4 x nscan4):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

incidenceAngleIndex (1-byte integer, array size: nchannel4 x nscan4):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

incidenceAngle(2,npixel,nscan)
sunGlintAngle(2,npixel,nscan)
incidenceAngleIndex(10,nscan)
Tc(10,npixel,nscan)

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

i = incidenceAngleIndex(channel,scan)
ia = incidenceAngle(i, pixel, scan)
sga = sunGlintAngle(i, pixel, scan)

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.

Values range from 0 to 100. Special values are defined as:
-99   Missing value

Tc (4-byte float, array size: nchannel4 x npixel4 x nscan4):
GPM Common Calibrated Brightness Temperature. The channels are:

<table>
<thead>
<tr>
<th>Frequency (GHz)</th>
<th>Polarization</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>165.5</td>
<td></td>
<td>quasi-horizontally-polarized TBs</td>
</tr>
<tr>
<td>183.31+-7</td>
<td></td>
<td>quasi-horizontally-polarized TBs</td>
</tr>
<tr>
<td>183.31+-4.5</td>
<td></td>
<td>quasi-horizontally-polarized TBs</td>
</tr>
<tr>
<td>183.31+-3</td>
<td></td>
<td>quasi-horizontally-polarized TBs</td>
</tr>
<tr>
<td>183.31+-1.8</td>
<td></td>
<td>quasi-horizontally-polarized TBs</td>
</tr>
<tr>
<td>183.31+-1</td>
<td></td>
<td>quasi-horizontally-polarized TBs</td>
</tr>
</tbody>
</table>

C Structure Header file:

```c
#define _TK_1CATMS_H_
#define _L1CATMS_S4_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[96];
    float Longitude[96];
    SCSTATUS SCstatus;
    signed char Quality[96];
    float incidenceAngle[96][1];
    signed char sunGlintAngle[96][1];
    signed char incidenceAngleIndex[6];
    float Tc[96][6];
} L1CATMS_S4;
```
typedef struct {
    SCANTIME ScanTime;
    float Latitude[96];
    float Longitude[96];
    SCSTATUS SCstatus;
    signed char Quality[96];
    float incidenceAngle[96][1];
    signed char sunGlintAngle[96][1];
    signed char incidenceAngleIndex[1];
    float Tc[96][1];
} L1CATMS_S3;

#endif

#ifndef _L1CATMS_S2_
define _L1CATMS_S2_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[96];
    float Longitude[96];
    SCSTATUS SCstatus;
    signed char Quality[96];
    float incidenceAngle[96][1];
    signed char sunGlintAngle[96][1];
    signed char incidenceAngleIndex[1];
    float Tc[96][1];
} L1CATMS_S2;
#endif

#ifndef _SCSTATUS_
define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
} SCSTATUS;
#endif
```c
float SCLongitude;
float SCAltitude;
double FractionalGranuleNumber;
} SCSTATUS;
#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L1CATMS_S1_
define _L1CATMS_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[96];
    float Longitude[96];
    SCSTATUS SCstatus;
signed char Quality[96];
    float incidenceAngle[96][1];
signed char sunGlintAngle[96][1];
signed char incidenceAngleIndex[1];
    float Tc[96][1];
} L1CATMS_S1;
#endif

#ifndef _L1CATMS_SWATHS_
define _L1CATMS_SWATHS_
```
typedef struct {
    L1CATMS_S1 S1;
    L1CATMS_S2 S2;
    L1CATMS_S3 S3;
    L1CATMS_S4 S4;
} L1CATMS_SWATHS;

#endif

#ifndef

Fortran Structure Header file:

STRUCTURE /L1CATMS_S4/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(96)
    REAL*4 Longitude(96)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(96)
    REAL*4 incidenceAngle(1,96)
    BYTE sunGlintAngle(1,96)
    BYTE incidenceAngleIndex(6)
    REAL*4 Tc(6,96)
END STRUCTURE

STRUCTURE /L1CATMS_S3/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(96)
    REAL*4 Longitude(96)
    RECORD /SCSTATUS/ SCstatus
    BYTE Quality(96)
    REAL*4 incidenceAngle(1,96)
    BYTE sunGlintAngle(1,96)
    BYTE incidenceAngleIndex(1)
    REAL*4 Tc(1,96)
END STRUCTURE

STRUCTURE /L1CATMS_S2/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(96)
    REAL*4 Longitude(96)
    RECORD /SCSTATUS/ SCstatus
END STRUCTURE

#endif
BYTE Quality(96)
REAL*4 incidenceAngle(1,96)
BYTE sunGlintAngle(1,96)
BYTE incidenceAngleIndex(1)
REAL*4 Tc(1,96)
END STRUCTURE

STRUCTURE /SCSTATUS/
  INTEGER*2 SCorientation
  REAL*4 SClatitude
  REAL*4 SClongitude
  REAL*4 SCaltitude
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1CATMS_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(96)
  REAL*4 Longitude(96)
  RECORD /SCSTATUS/ SCstatus
  BYTE Quality(96)
  REAL*4 incidenceAngle(1,96)
  BYTE sunGlintAngle(1,96)
  BYTE incidenceAngleIndex(1)
  REAL*4 Tc(1,96)
END STRUCTURE

STRUCTURE /L1CATMS_SWATHS/
  RECORD /L1CATMS_S1/ S1;
  RECORD /L1CATMS_S2/ S2;
  RECORD /L1CATMS_S3/ S3;
5.31 1CAMSUB - Common Calibrated Brightness Temperature

1CAMSUB contains common calibrated brightness temperature from the AMSU-B passive microwave instrument flown on the NOAA satellites. Swath S1 is the only swath and has 5 channels (89.0 +/- 0.9 GHz, 150.0 +/- 0.9 GHz, 183.31 +/- 1 GHz, 183.31 +/- 3 GHz, and 183.31 +/- 7 GHz) AMSU-B is very similar to MHS. The scan period is 2.667s.

RELATION BETWEEN THE SWATHS: S1 is the only swath, containing observations sampled 90 times along the scan.

KNOWN PROBLEMS OR ISSUES WITH REVISION 1 DATA: None.

Dimension definitions:

- `nscan1`: Number of Swath 1 scans in the granule.
- `nchannel1`: Number of Swath 1 channels.
- `npixel1`: Number of Swath 1 pixels in one scan.
- `nchUIA1`: Number of Swath S1 unique incidence angles.

Figure 414 through Figure 417 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.
5 STANDARD GPM PRODUCTS

Figure 415: Data Format Structure for 1CAMSUB, S1

ScanTime
- Year 2 bytes (Array: nscan1)
- Month 1 byte (Array: nscan1)
- DayOfMonth 1 byte (Array: nscan1)
- Hour 1 byte (Array: nscan1)
- Minute 1 byte (Array: nscan1)
- Second 1 byte (Array: nscan1)
- MilliSecond 2 bytes (Array: nscan1)
- DayOfYear 2 bytes (Array: nscan1)
- SecondOfDay 8 bytes (Array: nscan1)

Figure 416: Data Format Structure for 1CAMSUB, ScanTime

SCstatus
- SCorientation 2 bytes (Array: nscan1)
- SClatitude 4 bytes (Array: nscan1)
- SClongitude 4 bytes (Array: nscan1)
- SCaltitude 4 bytes (Array: nscan1)
- FractionalGranuleNumber 8 bytes (Array: nscan1)

Figure 417: Data Format Structure for 1CAMSUB, SCstatus
**NavigationRecord** (Metadata):  
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):  
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**XCALinfo** (Metadata):  
XCALinfo contains metadata required by 1C intercalibrated files. See Metadata for GPM Products for details.

**S1** (Swath)

**S1_SwathHeader** (Metadata):  
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**S1_IncidenceAngleIndex** (Metadata):  
IncidenceAngleIndex contains a list of indeces of the incidence angle array and sun glint angle array. See the description of the data array incidenceAngleIndex for details.

**ScanTime** (Group)  
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan1):  
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan1):  
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan1):  
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan1):  
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan1):  
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value
Second (1-byte integer, array size: nscan1):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan1):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan1):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan1):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel1 x nscan1):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel1 x nscan1):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group)

SCorientation (2-byte integer, array size: nscan1):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan1):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan1):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
SCAltitude (4-byte float, array size: nscan1):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan1):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

Quality (1-byte integer, array size: npixel1 x nscan1):
Quality of Tc in the swath.

GENERAL SPECIFICATIONS:
0 = Good data in all channels in the swath
gt 0 = Cautionary warning flags
    1-99 = Generic flags (all sensors)
    100-127 = Sensor specific flags
lt 0 = Major errors resulting in missing data
   -(1-98) = Generic flags (all sensors)
   -99 = Missing value (no quality information available)
   -(100-127) = Sensor specific flags

DETAILED SPECIFICATIONS:
1 = Possible sunGlint, 0 le sunGlintAngle lt 20
2 = Possible radio frequency interference
3 = Degraded geolocation data
4 = Data corrected for warm load intrusion

-1 = Data is missing from file or unreadable, missing scan
-2 = Invalid Tb or unphysical brightness temperature Tb lt 50 or Tb gt 350
-3 = Error in geolocation
-4 = Data is missing in 1 channel
-5 = Data is missing in multiple channels
-6 = Lat/Lon values are out of range
-7 = Non-normal status modes
-10 = Distance to its corresponding LF pixel exceeds 7Km
      threshold. used in L1C-R product only
-99 = Missing value (no quality information available)

-100 = data not useable in 89 GHz channel
-101 = data not useable in 150 GHz channel
-102 = data not useable in 183+/-1 GHz channel
-103 = data not useable in 183+/-3 GHz channel
-104 = data not useable in 183+/-7 GHz channel
-105 = data not useable in multiple channels

**incidenceAngle** (4-byte float, array size: nchUIA1 x npixel1 x nscan1):
Earth incidence angle, the angle of the satellite from the local zenith as seen at the pixel location on the earth. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**sunGlintAngle** (1-byte integer, array size: nchUIA1 x npixel1 x nscan1):
Sun glint angle. Angles greater than 127 degrees are set to 127. Values range from 0 to 127 degrees. Sun below horizon value is -88. Missing value is -99.

**incidenceAngleIndex** (1-byte integer, array size: nchannel1 x nscan1):

Index (1 based as in Fortran) of the incidence angle array corresponding to the channel. For example, if the swath has 10 channels and 2 unique incidence angles, then the dimensions in Fortran would be:

\[
\begin{align*}
\text{incidenceAngle}(2,\text{npixel},\text{nscan}) \\
\text{sunGlintAngle}(2,\text{npixel},\text{nscan}) \\
\text{incidenceAngleIndex}(10,\text{nscan}) \\
\text{Tc}(10,\text{npixel},\text{nscan})
\end{align*}
\]

The user would do the following to retrieve the angles for a given channel, pixel, and scan:

\[
\begin{align*}
i & = \text{incidenceAngleIndex}(\text{channel},\text{scan}) \\
i_a & = \text{incidenceAngle}(i,\text{pixel},\text{scan}) \\
s_ga & = \text{sunGlintAngle}(i,\text{pixel},\text{scan})
\end{align*}
\]

The incidenceAngleIndex is the same for every scan, but is repeated each scan for the convenience of users reading the data scan by scan. In addition, incidenceAngleIndex is located in metadata for the convenience of users wishing to read this information from metadata.
Values range from 0 to 100. Special values are defined as:
-99 Missing value

\( T_c \) (4-byte float, array size: \( n_{channel1 \times npixel1 \times n_{scan1}} \)):
GPM Common Calibrated Brightness Temperature. The channels are:

- 89.0 +/- 0.9 GHz TBs
- 150.0 +/- 0.9 GHz TBs
- 183.31 +/- 1 GHz TBs
- 183.31 +/- 3 GHz TBs
- 183.31 +/- 7 GHz TBs

C Structure Header file:

```c
#ifndef _TK_1CAMSUB_H_
define _TK_1CAMSUB_H_
#endif

#ifndef _SCSTATUS_
define _SCSTATUS_
#endif
typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
#endif

#ifndef _SCANTIME_
define _SCANTIME_
typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif
```
typedef struct {
    SCANTIME ScanTime;
    float Latitude[90];
    float Longitude[90];
    SCSTATUS SCstatus;
    signed char Quality[90];
    float incidenceAngle[90][1];
    signed char sunGlintAngle[90][1];
    signed char incidenceAngleIndex[5];
    float Tc[90][5];
} L1CAMSUB_S1;

typedef struct {
    L1CAMSUB_S1 S1;
} L1CAMSUB_SWATHS;

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude
    REAL*4 SCaltitude
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year
5.32 2AGPROFGMI - Radiometer Profiling

2AGPROFGMI, "Radiometer Profiling", generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2014. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 80 typical structures for each hydrometeor or heating profile. These vertical structures are referenced to as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written for each pixel.

Two products use the 2AGPROFGMI format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Dimension definitions:
nscan var Number of scans in the granule.

npixel 221 Number of pixels in each scan.

nspecies 5 Number of hydrometeor species. Species are defined in speciesDescription in the DataHeader group.

sddim 21 Number of characters in each species description.

ntemps 12 Number of profile temperature indeces. Indeces are defined in temperatureDescriptions in the DataHeader group.

nlyrs 28 Number of profiling layers. The top height of each layer is defined in hgtTopLayer in the DataHeader group.

nprf 80 Number of unique profiles for each species and 2 meter Temperature index.

Figure 418 through Figure 422 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**GprofInfo** (Metadata):
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

**GprofDHeadr** (Group)

speciesDescription (1-byte char, array size: sddim x nspecies):
Description of each species. Special values are defined as:
255 Missing value

hgtTopLayer (4-byte float, array size: nlyrs):
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ...
Figure 418: Data Format Structure for 2AGPROFGMI, Radiometer Profiling
continued from last figure

- convectivePrecipitation 4 bytes Array: npixel x nscan
- rainWaterPath 4 bytes Array: npixel x nscan
- cloudWaterPath 4 bytes Array: npixel x nscan
- iceWaterPath 4 bytes Array: npixel x nscan
- mostLikelyPrecipitation 4 bytes Array: npixel x nscan
- precip1stTertial 4 bytes Array: npixel x nscan
- precip2ndTertial 4 bytes Array: npixel x nscan
- profileTemp2mIndex 2 bytes Array: npixel x nscan
- profileNumber 2 bytes Array: nspecies x npixel x nscan
- profileScale 4 bytes Array: nspecies x npixel x nscan

Figure 419: Data Format Structure for 2AGPROFGMI, Radiometer Profiling

- speciesDescription 1 byte Array: sddim x nspecies
- hgtTopLayer 4 bytes Array: nlyrs
- temperatureDescriptions 4 bytes Array: ntemps
- clusterProfiles 4 bytes Array: nspecies x ntemps x nlyrs x nprf

Figure 420: Data Format Structure for 2AGPROFGMI, GprofDHeadr

- Year 2 bytes Array: nscan
- Month 1 byte Array: nscan
- DayOfMonth 1 byte Array: nscan
- Hour 1 byte Array: nscan
- Minute 1 byte Array: nscan
- Second 1 byte Array: nscan
- MilliSecond 2 bytes Array: nscan
- DayOfYear 2 bytes Array: nscan
- SecondOfYear 8 bytes Array: nscan

Figure 421: Data Format Structure for 2AGPROFGMI, ScanTime
SCstatus

- SCorientation 2 bytes Array: nscan
- SClatitude 4 bytes Array: nscan
- SClongitude 4 bytes Array: nscan
- SCaltitude 4 bytes Array: nscan
- FractionalGranuleNumber 8 bytes Array: nscan

Figure 422: Data Format Structure for 2AGPROFGMI, SCstatus

9.5, 10.0, 11.0, ... 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 18.0 km. Special values are defined as:
-9999.9 Missing value

temperatureDescriptions (4-byte float, array size: ntemps):
Temperature of 2 meter temperature indeces of clusterProfiles. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in C. Special values are defined as:
-9999.9 Missing value

clusterProfiles (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2 meter temperature index (12); vertical layers (28); and profile number (80). To recover values in a profile see the description below in the variable profileScale. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data.

Special values are defined as:
-9999.9 Missing value

S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value
Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCstatus (Group)
SCorientation (2-byte integer, array size: nscan):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

pixelStatus (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

0 : Valid pixel
1 : Invalid Latitude / Longitude
2 : Channel Tbs out of range
3 : Surface code / histogram mismatch
4 : Missing TCWV, T2m, or sfccode from preprocessor
5 : No Bayesian Solution
-99 : Missing value

qualityFlag (1-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 4).

Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution." Pixels can be set to 1 for the following reasons:
   - Sunglint is present, RFI, geolocate, warm load
     or other L1C 'positive value' quality warning flags.
   - All sea-ice covered surfaces.
   - All snow covered surfaces.
Sensor channels are missing, but not critical ones.

2: "Use pixel with extreme care over snow covered surface."
This is a special value for snow covered surfaces only. The pixel is set to 2 if the probability of precipitation is of poor quality or indeterminate. Use these pixels for climatological averaging of precipitation, but not for individual storm scale daily cases.

3: "Use with extreme caution." Pixels are set to 3 if they have channels missing critical to the retrieval, but the choice has been made to continue the retrieval for the pixel.

-99: Missing value

**L1CQualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.

0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
Negative: Copied from negative 1C Quality flag (GMI only)

**surfaceTypeIndex** (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

Codes include:
1: Ocean
2: Sea-Ice
3-7: Decreasing vegetation
8-11: Decreasing snow cover
12: Standing Water
13: Land/ocean or water Coast
14: Sea-ice edge
-99: Missing value

**totalColumnWaterVaporIndex** (1-byte integer, array size: npixel x nscan):
The integer total precipitable water used to select the correct database profiles. TotalColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof product the GANAL model is used. In the NRT Gprof product the JMAfcst model is used. Values range from 0 to 78 mm. Special values are defined as:

-99: Missing value

**CAPE** (2-byte integer, array size: npixel x nscan):
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range
from 1 to 5. Special values are defined as:
-9999 Missing value

**temp2mIndex** (2-byte integer, array size: npixel x nscan):
The 2 meter temperature Index used to select profiles in the database. Values are in K. Special values are defined as:
-9999 Missing value

**sunGlintAngle** (1-byte integer, array size: npixel x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. sunGlintAngle is the angular separation between the reflected satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values range from 0 to 127 degrees. Special values are defined as:
-88 Sun below horizon
-99 Missing

**probabilityOfPrecip** (1-byte integer, array size: npixel x nscan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
-99 Missing value

**spare2** (2-byte integer, array size: npixel x nscan):
Spare variable.

**surfacePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**frozenPrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous frozen precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined from Combined Profiles addition of snow and graupel in the lowest profile level with a rate of fall factor. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**convectivePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**rainWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0
(not -9999.9) was used to denote missing data. Values range from 0 to 3000 \( kg/m^2 \). Special values are defined as:
-9999.9 Missing value

**cloudWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 \( kg/m^2 \). Special values are defined as:
-9999.9 Missing value

**iceWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 \( kg/m^2 \). Special values are defined as:
-9999.9 Missing value

**mostLikelyPrecipitation** (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip1stTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip2ndTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values are defined as:
-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:
-9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description below. Values range from 1 to 80. Special values are defined as:
-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel, select your species, level, and profile2mTempIndex,
then use profileNumber and profileScale to obtain the value:

Where:

- \( S \) = species (1-5)
  - Species defined in speciesDescription
- \( T \) = profile2mTempIndex (1-12)
  - Temperatures defined in temperatureDescriptions
- \( L \) = profile level (1-28) Top of each level specified in hgtTopLayer
- \( P \) = profileNumber (1-80) for species \( S \)

In a Fortran program,
\[
P = \text{profileNumber}(S)
\]
Pixel Value = \( \text{profileScale}(S) \times \text{clusterProfiles}(S,T,L,P) \)

In a C program,
\[
P = \text{profileNumber}[(S-1)]
\]
Pixel Value = \( \text{profileScale}[S] \times \text{clusterProfiles}[P-1][L-1][T-1][S-1] \)

**C Structure Header file:**

```c
#ifndef _TK_2AGPROFGMI_H_
#define _TK_2AGPROFGMI_H_

#ifndef _SCSTATUS_
#define _SCSTATUS_

typedef struct {
  short SOrientation;
  float SLatitude;
  float SLongitude;
  float SCAltitude;
  double FractionalGranuleNumber;
} SCSTATUS;
#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
  short Year;
} SCANTIME;
#endif

#endif
```
signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
  double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L2AGPROFGMI_S1_
#define _L2AGPROFGMI_S1_

typedef struct {
  SCANTIME ScanTime;
  float Latitude[221];
  float Longitude[221];
  SCSTATUS SCstatus;
  signed char pixelStatus[221];
  signed char qualityFlag[221];
  signed char L1CqualityFlag[221];
  signed char surfaceTypeIndex[221];
  signed char totalColumnWaterVaporIndex[221];
  short CAPE[221];
  short temp2mIndex[221];
  signed char sunGlintAngle[221];
  signed char probabilityOfPrecip[221];
  short spare2[221];
  float surfacePrecipitation[221];
  float frozenPrecipitation[221];
  float convectivePrecipitation[221];
  float rainWaterPath[221];
  float cloudWaterPath[221];
  float iceWaterPath[221];
  float mostLikelyPrecipitation[221];
  float precip1stTertial[221];
  float precip2ndTertial[221];
  short profileTemp2mIndex[221];
  short profileNumber[221][5];
  float profileScale[221][5];
} L2AGPROFGMI_S1;
#endif

#ifndef _GPROFDHEADR_
define _GPROFDHEADR_

typedef struct {
    unsigned char speciesDescription[5][21];
    float hgtTopLayer[28];
    float temperatureDescriptions[12];
    float clusterProfiles[80][28][12][5];
} GPROFDHEADR;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
   INTEGER*2 SCorientation
   REAL*4 SClatitude
   REAL*4 SClongitude
   REAL*4 SCaltitude
   REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
   INTEGER*2 Year
   BYTE Month
   BYTE DayOfMonth
   BYTE Hour
   BYTE Minute
   BYTE Second
   INTEGER*2 MilliSecond
   INTEGER*2 DayOfYear
   REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFGMI_S1/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(221)
   REAL*4 Longitude(221)
5.33 2AGPROFTMI - Radiometer Profiling

2AGPROFTMI, "Radiometer Profiling", generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2014. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 100 typical structures for each hydrometeor or heating profile. These vertical structures are referenced to as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written
for each pixel.

Two products use this format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Dimension definitions:

- `nscan` var: Number of scans in the granule.
- `npixel` 208: Number of pixels in each scan.
- `nspecies` 5: Number of hydrometeor species. Species are defined in speciesDescription in the DataHeader group.
- `sddim` 21: Number of characters in each species description.
- `ntemps` 12: Number of profile temperature indeces. Indeces are defined in temperatureDescriptions in the DataHeader group.
- `nlyrs` 28: Number of profiling layers. The top height of each layer is defined in hgtTopLayer in the DataHeader group.
- `nprf` 80: Number of unique profiles for each species and 2 meter Temperature index.

Figure 423 through Figure 427 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**GprofInfo (Metadata):**
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

**GprofDHeadr (Group)**
Figure 423: Data Format Structure for 2AGPROFTMI, Radiometer Profiling
continued from last figure

- **convectivePrecipitation**: 4 bytes, Array: npixel x nscan
- **rainWaterPath**: 4 bytes, Array: npixel x nscan
- **cloudWaterPath**: 4 bytes, Array: npixel x nscan
- **iceWaterPath**: 4 bytes, Array: npixel x nscan
- **mostLikelyPrecipitation**: 4 bytes, Array: npixel x nscan
- **precip1stTertial**: 4 bytes, Array: npixel x nscan
- **precip2ndTertial**: 4 bytes, Array: npixel x nscan
- **profileTemp2mIndex**: 2 bytes, Array: npixel x nscan
- **profileNumber**: 2 bytes, Array: nspecies x npixel x nscan
- **profileScale**: 4 bytes, Array: nspecies x npixel x nscan

Figure 424: Data Format Structure for 2AGPROFTMI, Radiometer Profiling

- **speciesDescription**: 1 byte, Array: sddim x nspecies
- **hgtTopLayer**: 4 bytes, Array: nlyrs
- **temperatureDescriptions**: 4 bytes, Array: ntemps
- **clusterProfiles**: 4 bytes, Array: nspecies x ntemps x nlyrs x nprf

Figure 425: Data Format Structure for 2AGPROFTMI, GprofDHeadr

- **Year**: 2 bytes, Array: nscan
- **Month**: 1 byte, Array: nscan
- **DayOfMonth**: 1 byte, Array: nscan
- **Hour**: 1 byte, Array: nscan
- **Minute**: 1 byte, Array: nscan
- **Second**: 1 byte, Array: nscan
- **MilliSecond**: 2 bytes, Array: nscan
- **DayOfYear**: 2 bytes, Array: nscan
- **SecondOfDay**: 8 bytes, Array: nscan

Figure 426: Data Format Structure for 2AGPROFTMI, ScanTime
speciesDescription (1-byte char, array size: sddim x nspecies):
Description of each species. Special values are defined as:
255 Missing value

hgtTopLayer (4-byte float, array size: nlyrs):
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ..., 9.5, 10.0, 11.0, ..., 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 18.0 km. Special values are defined as:
-9999.9 Missing value

temperatureDescriptions (4-byte float, array size: ntemps):
Temperature of 2 meter temperature indeces of clusterProfiles. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in C. Special values are defined as:
-9999.9 Missing value

clusterProfiles (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2 meter temperature index (12); vertical layers (28); and profile number (80). To recover values in a profile see the description below in the variable profileScale. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data.

Special values are defined as:
-9999.9 Missing value

S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
  -9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
  -99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
  -99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
  -99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
  -99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
  -99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
  -9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
  -9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
  -9999.9 Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
  -9999.9 Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCstatus (Group)**

**SCorientation (2-byte integer, array size: nscan):**
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

**SClatitude (4-byte float, array size: nscan):**
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**SClongitude (4-byte float, array size: nscan):**
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCaltitude (4-byte float, array size: nscan):**
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

**FractionalGranuleNumber (8-byte float, array size: nscan):**
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**pixelStatus (1-byte integer, array size: npixel x nscan):**
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

0 : Valid pixel
1 : Invalid Latitude / Longitude
2 : Channel Tbs out of range
3 : Surface code / histogram mismatch
4 : Missing TCWV, T2m, or sfccode from preprocessor
5 : No Bayesian Solution
-99 : Missing value

**qualityFlag (1-byte integer, array size: npixel x nscan):**
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 4).
Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution." Pixels can be set to 1 for the following reasons:
   - Sunglint is present, RFI, geolocate, warm load
   - All sea-ice covered surfaces.
   - All snow covered surfaces.
   - Sensor channels are missing, but not critical ones.
2 : "Use pixel with extreme care over snow covered surface."
   This is a special value for snow covered surfaces only.
   The pixel is set to 2 if the probability of precipitation is of poor quality or indeterminate.
   Use these pixels for climatological averaging of precipitation, but not for individual storm scale daily cases.
3 : "Use with extreme caution." Pixels are set to 3 if they have channels missing critical to the retrieval,
   but the choice has been made to continue the retrieval for the pixel.
-99 : Missing value

**L1CqualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.

0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
   Negative: Copied from negative 1C Quality flag (GMI only)

**surfaceTypeIndex** (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

Codes include
1 : Ocean
2 : Sea-Ice
3-7 : Decreasing vegetation
8-11 : Decreasing snow cover
12 : Standing Water
13 : Land/ocean or water Coast
14 : Sea-ice edge
-99 : Missing value

**totalColumnWaterVaporIndex** (1-byte integer, array size: npixel x nscan):
The integer total precipitable water used to select the correct database profiles. Total-ColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable
Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof product the GANAL model is used. In the NRT Gprof product the JMAfcst model is used. Values range from 0 to 78 mm. Special values are defined as:

-99 Missing value

**CAPE** (2-byte integer, array size: npixel x nscan):
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range from 1 to 5. Special values are defined as:

-9999 Missing value

**temp2mIndex** (2-byte integer, array size: npixel x nscan):
The 2 meter temperature Index used to select profiles in the database. Values are in K. Special values are defined as:

-9999 Missing value

**sunGlintAngle** (1-byte integer, array size: npixel x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. sunGlintAngle is the angular separation between the reflected satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values range from 0 to 127 degrees. Special values are defined as:

-88 Sun below horizon
-99 Missing

**probabilityOfPrecip** (1-byte integer, array size: npixel x nscan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:

-99 Missing value

**spare2** (2-byte integer, array size: npixel x nscan):
Spare variable.

**surfacePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:

-9999.9 Missing value

**frozenPrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous frozen precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined from Combined Profiles addition of snow and graupule in the lowest profile level with a rate of fall factor. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:

-9999.9 Missing value

**convectivePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a
valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**rainWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

**cloudWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

**iceWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

**mostLikelyPrecipitation** (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip1stTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip2ndTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values are defined as:
-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:
-9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description
below. Values range from 1 to 80. Special values are defined as:
-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel,
select your species, level, and profile2mTempIndex,
then use profileNumber and profileScale
to obtain the value:

Where:
- **S** = species (1-5)
  Species defined in speciesDescription
- **T** = profile2mTempIndex (1-12)
  Temperatures defined in temperatureDescriptions
- **L** = profile level (1-28) Top of each level
  specified in hgtTopLayer
- **P** = profileNumber (1-80) for species **S**

In a Fortran program,
- \( P = \text{profileNumber}(S) \)
- Pixel Value = \( \text{profileScale}(S) \times \text{clusterProfiles}(S,T,L,P) \)

In a C program,
- \( P = \text{profileNumber}[S-1] \)
- Pixel Value = \( \text{profileScale}[S] \times \text{clusterProfiles}[P-1][L-1][T-1][S-1] \)

**C Structure Header file:**

```c
#ifndef _TK_2AGPROFTMI_H_
#define _TK_2AGPROFTMI_H_

#ifndef _SCSTATUS_
#define _SCSTATUS_

typedef struct { 
  short SCorientation;
  float SClatitude;
  float SClongitude;
  float SCaltitude;
  double FractionalGranuleNumber;
} SCSTATUS;
```
#ifdef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L2AGPROFTMI_S1_
#define _L2AGPROFTMI_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[208];
    float Longitude[208];
    SCSTATUS SCstatus;
    signed char pixelStatus[208];
    signed char qualityFlag[208];
    signed char LiCqualityFlag[208];
    signed char surfaceTypeIndex[208];
    signed char totalColumnWaterVaporIndex[208];
    short CAPE[208];
    short temp2mIndex[208];
    signed char sunGlintAngle[208];
    signed char probabilityOfPrecip[208];
    short spare2[208];
    float surfacePrecipitation[208];
    float frozenPrecipitation[208];
    float convectivePrecipitation[208];
    float rainWaterPath[208];
    float cloudWaterPath[208];
    float iceWaterPath[208];
}
float mostLikelyPrecipitation[208];
float precip1stTertial[208];
float precip2ndTertial[208];
short profileTemp2mIndex[208];
short profileNumber[208][5];
float profileScale[208][5];
} L2AGPROFTMI_S1;

#endif

#ifndef _GPROFDHEADR_
#define _GPROFDHEADR_

typedef struct {
  unsigned char speciesDescription[5][21];
  float hgtTopLayer[28];
  float temperatureDescriptions[12];
  float clusterProfiles[80][28][12][5];
} GPROFDHEADR;
#endif

#endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
  INTEGER*2 SCorientation
  REAL*4 SClatitude
  REAL*4 SClongitude
  REAL*4 SCaltitude
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFTMI_S1/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(208)
    REAL*4 Longitude(208)
    RECORD /SCSTATUS/ SCstatus
    BYTE pixelStatus(208)
    BYTE qualityFlag(208)
    BYTE LiCqualityFlag(208)
    BYTE surfaceTypeIndex(208)
    BYTE totalColumnWaterVaporIndex(208)
    INTEGER*2 CAPE(208)
    INTEGER*2 temp2mIndex(208)
    BYTE sunGlintAngle(208)
    BYTE probabilityOfPrecip(208)
    INTEGER*2 spare2(208)
    REAL*4 surfacePrecipitation(208)
    REAL*4 frozenPrecipitation(208)
    REAL*4 convectivePrecipitation(208)
    REAL*4 rainWaterPath(208)
    REAL*4 cloudWaterPath(208)
    REAL*4 iceWaterPath(208)
    REAL*4 mostLikelyPrecipitation(208)
    REAL*4 precip1stTertial(208)
    REAL*4 precip2ndTertial(208)
    INTEGER*2 profileTemp2mIndex(208)
    INTEGER*2 profileNumber(5,208)
    REAL*4 profileScale(5,208)
END STRUCTURE

STRUCTURE /GPROFDHEADR/
    CHARACTER speciesDescription(21,5)
    REAL*4 hgtTopLayer(28)
    REAL*4 temperatureDescriptions(12)
    REAL*4 clusterProfiles(5,12,28,80)
END STRUCTURE
5.34 2AGPROFSSMI - Radiometer Profiling

2AGPROFSSMI, ”Radiometer Profiling”, generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2017. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 80 typical structures for each hydrometeor or heating profile. These vertical structures are referenced to as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written for each pixel.

Two products use the 2AGPROFSSMI format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Dimension definitions:

- **nscan**: Number of scans in the granule.
- **npixel**: Number of pixels in each scan.
- **nspecies**: Number of hydrometeor species. Species are defined in speciesDescription in the DataHeader group.
- **sddim**: Number of characters in each species description.
- **ntemps**: Number of profile temperature indeces. Indeces are defined in temperatureDescriptions in the DataHeader group.
- **nlyrs**: Number of profiling layers. The top height of each layer is defined in hgtTopLayer in the DataHeader group.
- **nprf**: Number of unique profiles for each species and 2 meter Temperature index.

Figure 428 through Figure 432 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
Figure 428: Data Format Structure for 2A PROFSSMI, Radiometer Profiling
continued from last figure

```
S1
  convectivePrecipitation  4 bytes  Array: npixel x nscan
  rainWaterPath  4 bytes  Array: npixel x nscan
  cloudWaterPath  4 bytes  Array: npixel x nscan
  iceWaterPath  4 bytes  Array: npixel x nscan
  mostLikelyPrecipitation  4 bytes  Array: npixel x nscan
  precip1stTertial  4 bytes  Array: npixel x nscan
  precip2ndTertial  4 bytes  Array: npixel x nscan
  profileTemp2mIndex  2 bytes  Array: npixel x nscan
  profileNumber  2 bytes  Array: nspecies x npixel x nscan
  profileScale  4 bytes  Array: nspecies x npixel x nscan
```

Figure 429: Data Format Structure for 2AGPROFSSMI, Radiometer Profiling

```
GprofDHeadr
  speciesDescription  1 byte  Array: sddim x nspecies
  hgtTopLayer  4 bytes  Array: nlyrs
  temperatureDescriptions  4 bytes  Array: ntemps
  clusterProfiles  4 bytes  Array: nspecies x ntemps x nlyrs x nprf
```

Figure 430: Data Format Structure for 2AGPROFSSMI, GprofDHeadr

```
ScanTime
  Year  2 bytes  Array: nscan
  Month  1 byte  Array: nscan
  DayOfMonth  1 byte  Array: nscan
  Hour  1 byte  Array: nscan
  Minute  1 byte  Array: nscan
  Second  1 byte  Array: nscan
  MilliSecond  2 bytes  Array: nscan
  DayOfYear  2 bytes  Array: nscan
  SecondOfDay  8 bytes  Array: nscan
```

Figure 431: Data Format Structure for 2AGPROFSSMI, ScanTime
5.34 2AGPROFSSMI - Radiometer Profiling

Figure 432: Data Format Structure for 2AGPROFSSMI, SCstatus

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**GprofInfo (Metadata):**
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

**GprofDHeadr (Group)**

**speciesDescription** (1-byte char, array size: sddim x nspecies):
Description of each species. Special values are defined as:
- 255 Missing value

**hgtTopLayer** (4-byte float, array size: nlyrs):
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ... 9.5, 10.0, 11.0, ... 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 18.0 km. Special values are defined as:
- 9999.9 Missing value

**temperatureDescriptions** (4-byte float, array size: ntemps):
Temperature of 2 meter temperature indeces of clusterProfiles. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in C. Special values are defined as:
- 9999.9 Missing value

**clusterProfiles** (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2 meter temperature index (12); vertical layers (28); and profile number (80). To recover values in a profile see the description below in the variable profileScale. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data.

Special values are defined as:
- 9999.9 Missing value
S1 (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value
**Latitude** (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**Longitude** (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**SCstatus** (Group)

**SCorientation** (2-byte integer, array size: nscan):
The angle of the spacecraft vector \(v\) from the satellite forward direction of motion, measured clockwise facing down. The relationship of \(v\) to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999  Missing value

**SClatitude** (4-byte float, array size: nscan):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**SClongitude** (4-byte float, array size: nscan):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**SCaltitude** (4-byte float, array size: nscan):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9  Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**pixelStatus** (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

0 : Valid pixel
1 : Invalid Latitude / Longitude
2 : Channel Tbs out of range
3 : Surface code / histogram mismatch
4 : Missing TCWV, T2m, or sfccode from preprocessor
5 : No Bayesian Solution
-99 : Missing value

**qualityFlag** (1-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 4).

Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution." Pixels can be set to 1 for the following reasons:
   - Sunglint is present, RFI, geolocate, warm load
   - or other L1C 'positive value' quality warning flags.
   - All sea-ice covered surfaces.
   - All snow covered surfaces.
   - Sensor channels are missing, but not critical ones.
2 : "Use pixel with extreme care over snow covered surface."
   This is a special value for snow covered surfaces only.
   The pixel is set to 2 if the probability of precipitation is of poor quality or indeterminate.
   Use these pixels for climatological averaging of precipitation, but not for individual storm scale daily cases.
3 : "Use with extreme caution." Pixels are set to 3 if they have channels missing critical to the retrieval,
   but the choice has been made to continue the retrieval for the pixel.
-99 : Missing value

**L1CqualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.

0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
Negative: Copied from negative 1C Quality flag (GMI only)

**surfaceTypeIndex** (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

Codes include
1 : Ocean
2 : Sea-Ice
3-7 : Decreasing vegetation
8-11 : Decreasing snow cover
12 : Standing Water
13 : Land/ocean or water Coast
14 : Sea-ice edge
-99 : Missing value

**totalColumnWaterVaporIndex** (1-byte integer, array size: npixel x nscan):
The integer total precipitable water used to select the correct database profiles. Total-ColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof product the GANAL model is used. In the NRT Gprof product the JMAfcst model is used. Values range from 0 to 78 mm. Special values are defined as:
-99 Missing value

**CAPE** (2-byte integer, array size: npixel x nscan):
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range from 1 to 5. Special values are defined as:
-9999 Missing value

**temp2mIndex** (2-byte integer, array size: npixel x nscan):
The 2 meter temperature Index used to select profiles in the database. Values are in K. Special values are defined as:
-9999 Missing value

**sunGlintAngle** (1-byte integer, array size: npixel x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. sunGlintAngle is the angular separation between the reflected satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values range from 0 to 127 degrees. Special values are defined as:
-88 Sun below horizon
-99 Missing

**probabilityOfPrecip** (1-byte integer, array size: npixel x nscan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
-99 Missing value

**spare2** (2-byte integer, array size: npixel x nscan):
Spare variable.

**surfacePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value
frozenPrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous frozen precipitation rate at the surface. Check pixelStatus for a valid
retrieval. Defined from Combined Profiles addition of snow and graupel in the lowest
profile level with a rate of fall factor. NOTE: In V05 -9999.0 (not -9999.9) was used to
denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

convectivePrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a
valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0
(not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are
defined as:
-9999.9 Missing value

rainWaterPath (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0
(not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m².
Special values are defined as:
-9999.9 Missing value

cloudWaterPath (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05
-9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000
kg/m². Special values are defined as:
-9999.9 Missing value

iceWaterPath (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not
-9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special
values are defined as:
-9999.9 Missing value

mostLikelyPrecipitation (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval.
NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in
mm/hr. Special values are defined as:
-9999.9 Missing value

precip1stTertial (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE:
In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr.
Special values are defined as:
-9999.9 Missing value

precip2ndTertial (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values
are defined as:
-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:
-9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description below. Values range from 1 to 80. Special values are defined as:
-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel, select your species, level, and profile2mTempIndex, then use profileNumber and profileScale to obtain the value:

Where:

S = species (1-5)
Species defined in speciesDescription
T = profile2mTempIndex (1-12)
Temperatures defined in temperatureDescriptions
L = profile level (1-28) Top of each level specified in hgtTopLayer
P = profileNumber (1-80) for species S

In a Fortran program,
P = profileNumber(S)
Pixel Value = profileScale(S) * clusterProfiles(S,T,L,P)

In a C program,
P = profileNumber[S-1]
Pixel Value = profileScale[S] * clusterProfiles[P-1][L-1][T-1][S-1]

**C Structure Header file:**

```c
#ifndef _TK_2AGPROFSSMI_H_
#define _TK_2AGPROFSSMI_H_

#ifndef _SCSTATUS_
#define _SCSTATUS_

#endif
#define _SCSTATUS_
```


typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;

#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L2AGPROFSSMI_S1_
define _L2AGPROFSSMI_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[128];
    float Longitude[128];
    SCSTATUS SCstatus;
    signed char pixelStatus[128];
    signed char qualityFlag[128];
    signed char L1CqualityFlag[128];
    signed char surfaceTypeIndex[128];
    signed char totalColumnWaterVaporIndex[128];
    short CAPE[128];
    short temp2mIndex[128];
}
signed char sunGlintAngle[128];
signed char probabilityOfPrecip[128];
short spare2[128];
float surfacePrecipitation[128];
float frozenPrecipitation[128];
float convectivePrecipitation[128];
float rainWaterPath[128];
float cloudWaterPath[128];
float iceWaterPath[128];
float mostLikelyPrecipitation[128];
float precip1stTertiary[128];
float precip2ndTertiary[128];
short profileTemp2mIndex[128];
short profileNumber[128][5];
float profileScale[128][5];
}
L2AGPROFSSMI_S1;

#endif

#ifndef _GPROFDHEADR_
#define _GPROFDHEADR_

typedef struct {
    unsigned char speciesDescription[5][21];
    float hgtTopLayer[28];
    float temperatureDescriptions[12];
    float clusterProfiles[80][28][12][5];
} GPROFDHEADR;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude
    REAL*4 SCaltitude
    REAL*8 FractionalGranuleNumber
END STRUCTURE
STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFSSMI_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(128)
  REAL*4 Longitude(128)
  RECORD /SCSTATUS/ SCstatus
  BYTE pixelStatus(128)
  BYTE qualityFlag(128)
  BYTE L1CqualityFlag(128)
  BYTE surfaceTypeIndex(128)
  INTEGER*2 CAPE(128)
  INTEGER*2 temp2mIndex(128)
  BYTE sunGlintAngle(128)
  BYTE probabilityOfPrecip(128)
  INTEGER*2 spare2(128)
  REAL*4 surfacePrecipitation(128)
  REAL*4 frozenPrecipitation(128)
  REAL*4 convectivePrecipitation(128)
  REAL*4 rainWaterPath(128)
  REAL*4 cloudWaterPath(128)
  REAL*4 iceWaterPath(128)
  REAL*4 mostLikelyPrecipitation(128)
  REAL*4 precip1stTertial(128)
  REAL*4 precip2ndTertial(128)
  INTEGER*2 profileTemp2mIndex(128)
  INTEGER*2 profileNumber(5,128)
  REAL*4 profileScale(5,128)
END STRUCTURE

STRUCTURE /GPROFDHEADR/
  CHARACTER speciesDescription(21,5)
5.35 2AGPROFSSMIS - Radiometer Profiling

2AGPROFSSMIS, "Radiometer Profiling", generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2014. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 100 typical structures for each hydrometeor or heating profile. These vertical structures are referenced to as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written for each pixel.

Two products use this format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Dimension definitions:

- nscan var Number of scans in the granule.
- npixel 180 Number of pixels in each scan.
- nspecies 5 Number of hydrometeor species. Species are defined in speciesDescription in the DataHeader group.
- sddim 21 Number of characters in each species description.
- ntemps 12 Number of profile temperature indeces. Indeces are defined in temperatureDescriptions in the DataHeader group.
- nlyrs 28 Number of profiling layers. The top height of each layer is defined in hgtTopLayer in the DataHeader group.
- nprf 80 Number of unique profiles for each species and 2 meter Temperature index.

Figure 433 through Figure 437 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information.
Figure 433: Data Format Structure for 2AGPROFSSMIS, Radiometer Profiling
continued from last figure

- **convectivePrecipitation**: 4 bytes, Array: npixel x nscan
- **rainWaterPath**: 4 bytes, Array: npixel x nscan
- **cloudWaterPath**: 4 bytes, Array: npixel x nscan
- **iceWaterPath**: 4 bytes, Array: npixel x nscan
- **mostLikelyPrecipitation**: 4 bytes, Array: npixel x nscan
- **precip1stTertial**: 4 bytes, Array: npixel x nscan
- **precip2ndTertial**: 4 bytes, Array: npixel x nscan
- **profileTemp2mIndex**: 2 bytes, Array: npixel x nscan
- **profileNumber**: 2 bytes, Array: nspecies x npixel x nscan
- **profileScale**: 4 bytes, Array: nspecies x npixel x nscan

Figure 434: Data Format Structure for 2AGPROFSSMIS, Radiometer Profiling

- **speciesDescription**: 1 byte, Array: sddim x nspecies
- **hgtTopLayer**: 4 bytes, Array: nlyrs
- **temperatureDescriptions**: 4 bytes, Array: ntemps
- **clusterProfiles**: 4 bytes, Array: nspecies x ntemps x nlyrs x nprf

Figure 435: Data Format Structure for 2AGPROFSSMIS, GprofDHeadr

- **Year**: 2 bytes, Array: nscan
- **Month**: 1 byte, Array: nscan
- **DayOfMonth**: 1 byte, Array: nscan
- **Hour**: 1 byte, Array: nscan
- **Minute**: 1 byte, Array: nscan
- **Second**: 1 byte, Array: nscan
- **MilliSecond**: 2 bytes, Array: nscan
- **DayOfYear**: 2 bytes, Array: nscan
- **SecondOfDay**: 8 bytes, Array: nscan

Figure 436: Data Format Structure for 2AGPROFSSMIS, ScanTime
separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**GprofInfo** (Metadata):
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

**GprofDHeadr** (Group)

**speciesDescription** (1-byte char, array size: sddim x nspecies):
Description of each species. Special values are defined as:
255 Missing value

**hgtTopLayer** (4-byte float, array size: nlyrs):
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ... 9.5, 10.0, 11.0, ... 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 18.0 km. Special values are defined as:
-9999.9 Missing value

**temperatureDescriptions** (4-byte float, array size: ntemps):
Temperature of 2 meter temperature indices of clusterProfiles. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in C. Special values are defined as:
-9999.9 Missing value

**clusterProfiles** (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2
meter temperature index (12); vertical layers (28); and profile number (80). To recover values in a profile see the description below in the variable profileScale. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data.

Special values are defined as:
-9999.9 Missing value

S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value
**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime.sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCstatus** (Group)

**SCorientation** (2-byte integer, array size: nscan):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

**SClatitude** (4-byte float, array size: nscan):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**SClongitude** (4-byte float, array size: nscan):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCaltitude** (4-byte float, array size: nscan):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**pixelStatus** (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

- 0 : Valid pixel
- 1 : Invalid Latitude / Longitude
- 2 : Channel Tbs out of range
- 3 : Surface code / histogram mismatch
- 4 : Missing TCWV, T2m, or sfccode from preprocessor
- 5 : No Bayesian Solution
- -99 : Missing value

**qualityFlag** (1-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 4).

Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution." Pixels can be set to 1 for the following reasons:
   - Sunglint is present, RFI, geolocate, warm load
     or other L1C 'positive value' quality warning flags.
   - All sea-ice covered surfaces.
   - All snow covered surfaces.
   - Sensor channels are missing, but not critical ones.
2 : "Use pixel with extreme care over snow covered surface."
   This is a special value for snow covered surfaces only.
   The pixel is set to 2 if the probability of precipitation
   is of poor quality or indeterminate. Use these pixels
   for climatological averaging of precipitation, but not
   for individual storm scale daily cases.
3 : "Use with extreme caution." Pixels are set to 3 if
   they have channels missing critical to the retrieval,
   but the choice has been made to continue the retrieval
   for the pixel.
-99 : Missing value

**L1CqualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.

0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
Negative: Copied from negative 1C Quality flag (GMI only)
surfaceTypeIndex (1-byte integer, array size: npixel x nsan):  
Indicates the type of surface (Range 0 - 99).

Codes include:
1 : Ocean
2 : Sea-Ice
3-7 : Decreasing vegetation
8-11 : Decreasing snow cover
12 : Standing Water
13 : Land/ocean or water Coast
14 : Sea-ice edge
-99 : Missing value

totalColumnWaterVaporIndex (1-byte integer, array size: npixel x nsan):
The integer total precipitable water used to select the correct database profiles. Total-ColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof product the GANAL model is used. In the NRT Gprof product the JMAfcst model is used. Values range from 0 to 78 mm. Special values are defined as:
-99 : Missing value

CAPE (2-byte integer, array size: npixel x nsan):
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range from 1 to 5. Special values are defined as:
-9999 : Missing value

temp2mIndex (2-byte integer, array size: npixel x nsan):
The 2 meter temperature Index used to select profiles in the database. Values are in K. Special values are defined as:
-9999 : Missing value

sunGlintAngle (1-byte integer, array size: npixel x nsan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. sunGlintAngle is the angular separation between the reflected satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values range from 0 to 127 degrees. Special values are defined as:
-88 : Sun below horizon
-99 : Missing

probabilityOfPrecip (1-byte integer, array size: npixel x nsan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
-99 : Missing value
**spare2** (2-byte integer, array size: npixel x nscan):
Spare variable.

**surfacePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval.
NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**frozenPrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous frozen precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined from Combined Profiles addition of snow and graupel in the lowest profile level with a rate of fall factor. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**convectivePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**rainWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

**cloudWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

**iceWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

**mostLikelyPrecipitation** (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval.
NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip1stTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE:
In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:

-9999.9 Missing value

**precip2ndTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution.

NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values are defined as:

-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:

-9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description below. Values range from 1 to 80. Special values are defined as:

-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel, select your species, level, and profile2mTempIndex, then use profileNumber and profileScale to obtain the value:

Where:

S = species (1-5)
   Species defined in speciesDescription
T = profile2mTempIndex (1-12)
   Temperatures defined in temperatureDescriptions
L = profile level (1-28) Top of each level specified in hgtTopLayer
P = profileNumber (1-80) for species S

In a Fortran program,

P = profileNumber(S)
Pixel Value = profileScale(S) * clusterProfiles(S,T,L,P)

In a C program,

P = profileNumber[S-1]
Pixel Value = profileScale[S] * clusterProfiles[P-1][L-1][T-1][S-1]

C Structure Header file:
#ifndef _TK_2AGPROFSSMIS_H_
#define _TK_2AGPROFSSMIS_H_

#ifndef _SCSTATUS_
#define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L2AGPROFSSMIS_S1_
#define _L2AGPROFSSMIS_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[180];
    float Longitude[180];
    SCSTATUS SCstatus;
    signed char pixelStatus[180];
} _L2AGPROFSSMIS_S1_;
signed char qualityFlag[180];
signed char L1CqualityFlag[180];
signed char surfaceTypeIndex[180];
signed char totalColumnWaterVaporIndex[180];
short CAPE[180];
short temp2mIndex[180];
signed char sunGlintAngle[180];
signed char probabilityOfPrecip[180];
short spare2[180];
float surfacePrecipitation[180];
float frozenPrecipitation[180];
float convectivePrecipitation[180];
float rainWaterPath[180];
float cloudWaterPath[180];
float iceWaterPath[180];
float mostLikelyPrecipitation[180];
float precip1stTertiary[180];
float precip2ndTertiary[180];
short profileTemp2mIndex[180];
short profileNumber[180][5];
float profileScale[180][5];
} L2AGPROFSSMIS_S1;
#endif

#ifndef _GPROFDHEADR_
define _GPROFDHEADR_

typedef struct {
   unsigned char speciesDescription[5][21];
   float hgtTopLayer[28];
   float temperatureDescriptions[12];
   float clusterProfiles[80][28][12][5];
} GPROFDHEADR;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
   INTEGER*2 SCoorientation
REAL*4 SClatitude
REAL*4 SClongitude
REAL*4 SCaltitude
REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFSSMIS_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(180)
  REAL*4 Longitude(180)
  RECORD /SCSTATUS/ SCstatus
  BYTE pixelStatus(180)
  BYTE qualityFlag(180)
  BYTE L1CqualityFlag(180)
  BYTE surfaceTypeIndex(180)
  INTEGER*2 CAPE(180)
  INTEGER*2 temp2mIndex(180)
  BYTE sunGlintAngle(180)
  BYTE probabilityOfPrecip(180)
  INTEGER*2 spare2(180)
  REAL*4 surfacePrecipitation(180)
  REAL*4 frozenPrecipitation(180)
  REAL*4 convectivePrecipitation(180)
  REAL*4 rainWaterPath(180)
  REAL*4 cloudWaterPath(180)
  REAL*4 iceWaterPath(180)
  REAL*4 mostLikelyPrecipitation(180)
  REAL*4 precip1stTertial(180)
  REAL*4 precip2ndTertial(180)
  INTEGER*2 profileTemp2mIndex(180)
INTEGER*2 profileNumber(5,180)
REAL*4 profileScale(5,180)
END STRUCTURE

STRUCTURE /GPROFDHEADR/
CHARACTER speciesDescription(21,5)
REAL*4 hgtTopLayer(28)
REAL*4 temperatureDescriptions(12)
REAL*4 clusterProfiles(5,12,28,80)
END STRUCTURE

5.36 2AGPROFAMSRE - Radiometer Profiling

2AGPROFAMSRE, "Radiometer Profiling", generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2014. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 100 typical structures for each hydrometeor or heating profile. These vertical structures are referenced to as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written for each pixel.

Two products use this format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Dimension definitions:

- nscan: Number of scans in the granule.
- npixel: Number of pixels in each scan.
- nspecies: Number of hydrometeor species. Species are defined in speciesDescription in the DataHeader group.
- sddim: Number of characters in each species description.
- ntemps: Number of profile temperature indexes. Indexes are defined in temperatureDescriptions in the DataHeader group.
- nlyrs: Number of profiling layers. The top height of each layer is defined in hgtTopLayer in the DataHeader group.
- nprf: Number of unique profiles for each species and 2 meter Temperature index.

Figure 438 through Figure 442 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 438: Data Format Structure for 2AGPROFAMSRE, Radiometer Profiling

continued on next figure

•

•
continued from last figure

- convectivePrecipitation: 4 bytes, Array: npixel x nscan
- rainWaterPath: 4 bytes, Array: npixel x nscan
- cloudWaterPath: 4 bytes, Array: npixel x nscan
- iceWaterPath: 4 bytes, Array: npixel x nscan
- mostLikelyPrecipitation: 4 bytes, Array: npixel x nscan
- precip1stTertial: 4 bytes, Array: npixel x nscan
- precip2ndTertial: 4 bytes, Array: npixel x nscan
- profileTemp2mIndex: 2 bytes, Array: npixel x nscan
- profileNumber: 2 bytes, Array: nspecies x npixel x nscan
- profileScale: 4 bytes, Array: nspecies x npixel x nscan

Figure 439: Data Format Structure for 2AGPROFAMSRE, Radiometer Profiling

- speciesDescription: 1 byte, Array: sddim x nspecies
- hgtTopLayer: 4 bytes, Array: nlyrs
- temperatureDescriptions: 4 bytes, Array: ntemps
- clusterProfiles: 4 bytes, Array: nspecies x ntemps x nlyrs x nprf

Figure 440: Data Format Structure for 2AGPROFAMSRE, GprofDHeadr

- Year: 2 bytes, Array: nscan
- Month: 1 byte, Array: nscan
- DayOfMonth: 1 byte, Array: nscan
- Hour: 1 byte, Array: nscan
- Minute: 1 byte, Array: nscan
- Second: 1 byte, Array: nscan
- MilliSecond: 2 bytes, Array: nscan
- DayOfYear: 2 bytes, Array: nscan
- SecondOfDay: 8 bytes, Array: nscan

Figure 441: Data Format Structure for 2AGPROFAMSRE, ScanTime
**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**GprofInfo (Metadata):**
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

---

**GprofDHeadr (Group)**

**speciesDescription** (1-byte char, array size: sddim x nspecies):
Description of each species. Special values are defined as:

   255   Missing value

**hgtTopLayer** (4-byte float, array size: nlyrs):
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ..., 9.5, 10.0, 11.0, ..., 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 18.0 km. Special values are defined as:

   -9999.9   Missing value
temperatureDescriptions (4-byte float, array size: ntemps):
Temperature of 2 meter temperature indeces of clusterProfiles. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in C. Special values are defined as:
-9999.9 Missing value

clusterProfiles (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2 meter temperature index (12); vertical layers (28); and profile number (80). To recover values in a profile see the description below in the variable profileScale. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data.

Special values are defined as:
-9999.9 Missing value

S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value
**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
- 99  Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
- 9999  Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
- 9999  Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
- 9999.9  Missing value

**Latitude** (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
- 9999.9  Missing value

**Longitude** (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9  Missing value

**SCstatus** (Group)

**SCorientation** (2-byte integer, array size: nscan):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
- 9999  Missing value

**SClatitude** (4-byte float, array size: nscan):
Values range from -90 to 90 degrees. Special values are defined as:
- 9999.9  Missing value

**SClongitude** (4-byte float, array size: nscan):
Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9  Missing value
SCaltitude (4-byte float, array size: nscan):
Values range from 0 to 1000 km. Special values are defined as:
- 9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
- 9999.9 Missing value

pixelStatus (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

0 : Valid pixel
1 : Invalid Latitude / Longitude
2 : Channel Tbs out of range
3 : Surface code / histogram mismatch
4 : Missing TCWV, T2m, or sfccode from preprocessor
5 : No Bayesian Solution
-99 : Missing value

qualityFlag (1-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 4).

Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution." Pixels can be set to 1 for the following reasons:
   - Sunglint is present, RFI, geolocate, warm load
     or other L1C 'positive value' quality warning flags.
   - All sea-ice covered surfaces.
   - All snow covered surfaces.
   - Sensor channels are missing, but not critical ones.
2 : "Use pixel with extreme care over snow covered surface."
   This is a special value for snow covered surfaces only. The
   pixel is set to 2 if the probability of precipitation
   is of poor quality or indeterminate. Use these pixels
   for climatological averaging of precipitation, but not
   for individual storm scale daily cases.
3 : "Use with extreme caution." Pixels are set to 3 if
   they have channels missing critical to the retrieval,
   but the choice has been made to continue the retrieval
   for the pixel.
-99 : Missing value
**L1CqualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.

0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
Negative: Copied from negative 1C Quality flag (GMI only)

**surfaceTypeIndex** (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

- Codes include:
  1 : Ocean
  2 : Sea-Ice
  3-7 : Decreasing vegetation
  8-11 : Decreasing snow cover
  12 : Standing Water
  13 : Land/ocean or water Coast
  14 : Sea-ice edge
  -99 : Missing value

**totalColumnWaterVaporIndex** (1-byte integer, array size: npixel x nscan):
The integer total precipitable water used to select the correct database profiles. TotalColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof product the GANAL model is used. In the NRT Gprof product the JMAfcst model is used. Values range from 0 to 78 mm. Special values are defined as:
-99 Missing value

**CAPE** (2-byte integer, array size: npixel x nscan):
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range from 1 to 5. Special values are defined as:
-9999 Missing value

**temp2mIndex** (2-byte integer, array size: npixel x nscan):
The 2 meter temperature Index used to select profiles in the database. Values are in K. Special values are defined as:
-9999 Missing value

**sunGlintAngle** (1-byte integer, array size: npixel x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. sunGlintAngle is the angular separation between the reflected satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values
range from 0 to 127 degrees. Special values are defined as:
-88 Sun below horizon
-99 Missing

**probabilityOfPrecip** (1-byte integer, array size: npixel x nscan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
-99 Missing value

**spare2** (2-byte integer, array size: npixel x nscan):
Spare variable.

**surfacePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**frozenPrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous frozen precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined from Combined Profiles addition of snow and graupel in the lowest profile level with a rate of fall factor. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**convectivePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**rainWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

**cloudWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

**iceWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special
values are defined as:
-9999.9 Missing value

**mostLikelyPrecipitation** (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval.
NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip1stTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE:
In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip2ndTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution.
NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values are defined as:
-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:
-9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description below. Values range from 1 to 80. Special values are defined as:
-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel, select your species, level, and profile2mTempIndex, then use profileNumber and profileScale to obtain the value:

Where:
S = species (1-5)  
   Species defined in speciesDescription  
T = profile2mTempIndex (1-12)  
   Temperatures defined in temperatureDescriptions  
L = profile level (1-28) Top of each level  
   specified in hgtTopLayer  
P = profileNumber (1-80) for species S
In a Fortran program,
\[ P = \text{profileNumber}(S) \]
\[ \text{Pixel Value} = \text{profileScale}(S) \times \text{clusterProfiles}(S,T,L,P) \]

In a C program,
\[ P = \text{profileNumber}[S-1] \]
\[ \text{Pixel Value} = \text{profileScale}[S] \times \text{clusterProfiles}[P-1][L-1][T-1][S-1] \]

**C Structure Header file:**

```c
#ifndef _TK_2AGPROFAMSRE_H_
define _TK_2AGPROFAMSRE_H_

#ifndef _SCSTATUS_
define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
#endif

define _SCSTATUS_

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

define _SCANTIME_
```

```
#ifndef _L2AGPROFAMSRE_S1_
define _L2AGPROFAMSRE_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[486];
    float Longitude[486];
    SCSTATUS SCstatus;
    signed char pixelStatus[486];
    signed char qualityFlag[486];
    signed char L1CqualityFlag[486];
    signed char surfaceTypeIndex[486];
    signed char totalColumnWaterVaporIndex[486];
    short CAPE[486];
    short temp2mIndex[486];
    signed char sunGlintAngle[486];
    signed char probabilityOfPrecip[486];
    short spare2[486];
    float surfacePrecipitation[486];
    float frozenPrecipitation[486];
    float convectivePrecipitation[486];
    float rainWaterPath[486];
    float cloudWaterPath[486];
    float iceWaterPath[486];
    float mostLikelyPrecipitation[486];
    float precip1stTertial[486];
    float precip2ndTertial[486];
    short profileTemp2mIndex[486];
    short profileNumber[486][5];
    float profileScale[486][5];
} L2AGPROFAMSRE_S1;

#endif

#elif _GPROFDHEADR_
define _GPROFDHEADR_

typedef struct {
    unsigned char speciesDescription[5][21];
    float hgtTopLayer[28];
    float temperatureDescriptions[12];
    float clusterProfiles[80][28][12][5];
} GPROFDHEADR;
Fortran Structure Header file:

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude
    REAL*4 SCaltitude
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFAMSRE_S1/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(486)
    REAL*4 Longitude(486)
    RECORD /SCSTATUS/ SCstatus
    BYTE pixelStatus(486)
    BYTE qualityFlag(486)
    BYTE L1CqualityFlag(486)
    BYTE surfaceTypeIndex(486)
    INTEGER*2 CAPE(486)
    INTEGER*2 temp2mIndex(486)
    BYTE sunGlintAngle(486)
    BYTE probabilityOfPrecip(486)
    INTEGER*2 spare2(486)
    REAL*4 surfacePrecipitation(486)
REAL*4 frozenPrecipitation(486)
REAL*4 convectivePrecipitation(486)
REAL*4 rainWaterPath(486)
REAL*4 cloudWaterPath(486)
REAL*4 iceWaterPath(486)
REAL*4 mostLikelyPrecipitation(486)
REAL*4 precip1stTertial(486)
REAL*4 precip2ndTertial(486)
INTEGER*2 profileTemp2mIndex(486)
INTEGER*2 profileNumber(5,486)
REAL*4 profileScale(5,486)
END STRUCTURE

STRUCTURE /GPROFDHEADR/
  CHARACTER speciesDescription(21,5)
  REAL*4 hgtTopLayer(28)
  REAL*4 temperatureDescriptions(12)
  REAL*4 clusterProfiles(5,12,28,80)
END STRUCTURE

5.37 2AGPROFAMSR2 - Radiometer Profiling

2AGPROFAMSR2, "Radiometer Profiling", generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2014. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 100 typical structures for each hydrometeor or heating profile. These vertical structures are referenced to as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written for each pixel.

Two products use this format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Dimension definitions:
nscan  var  Number of scans in the granule.
npixel  486  Number of pixels in each scan.
nspecies  5  Number of hydrometeor species. Species are defined in speciesDescription in the DataHeader group.
sddim  21  Number of characters in each species description.
ntemps  12  Number of profile temperature indeces. Indeces are defined in temperatureDescriptions in the DataHeader group.
nllyrs  28  Number of profiling layers. The top height of each layer is defined in hgtTopLayer in the DataHeader group.
nprf  80  Number of unique profiles for each species and 2 meter Temperature index.

Figure 443 through Figure 447 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

GprofInfo (Metadata):
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

GprofDHeadr (Group)

speciesDescription (1-byte char, array size: sddim x nspecies):
Description of each species. Special values are defined as:
255  Missing value

hgtTopLayer (4-byte float, array size: nlyrs):
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ...
Figure 443: Data Format Structure for 2AGPROFAMSR2, Radiometer Profiling
continued from last figure

- convectivePrecipitation 4 bytes  Array: npixel x nscan
- rainWaterPath 4 bytes  Array: npixel x nscan
- cloudWaterPath 4 bytes  Array: npixel x nscan
- iceWaterPath 4 bytes  Array: npixel x nscan
- mostLikelyPrecipitation 4 bytes  Array: npixel x nscan
- precip1stTertial 4 bytes  Array: npixel x nscan
- precip2ndTertial 4 bytes  Array: npixel x nscan
- profileTemp2mIndex 2 bytes  Array: npixel x nscan
- profileNumber 2 bytes  Array: nspecies x npixel x nscan
- profileScale 4 bytes  Array: nspecies x npixel x nscan

Figure 444: Data Format Structure for 2AGPROFAMS R2, Radiometer Profiling

- speciesDescription 1 byte  Array: sddim x nspecies
- hgtTopLayer 4 bytes  Array: nlyrs
- temperatureDescriptions 4 bytes  Array: ntemps
- clusterProfiles 4 bytes  Array: nspecies x ntemps x nlyrs x nprf

Figure 445: Data Format Structure for 2AGPROFAMS R2, GprofDHeadr

- Year 2 bytes  Array: nscan
- Month 1 byte  Array: nscan
- DayOfMonth 1 byte  Array: nscan
- Hour 1 byte  Array: nscan
- Minute 1 byte  Array: nscan
- Second 1 byte  Array: nscan
- MilliSecond 2 bytes  Array: nscan
- DayOfYear 2 bytes  Array: nscan
- SecondOfWeek 8 bytes  Array: nscan

Figure 446: Data Format Structure for 2AGPROFAMS R2, ScanTime
Data Format Structure for 2AGPROFAMSR2, SCstatus

9.5, 10.0, 11.0, ... 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 18.0 km. Special values are defined as:
-9999.9 Missing value

**temperatureDescriptions** (4-byte float, array size: ntemps):
Temperature of 2 meter temperature indeces of clusterProfiles. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in C. Special values are defined as:
-9999.9 Missing value

**clusterProfiles** (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2 meter temperature index (12); vertical layers (28); and profile number (80). To recover values in a profile see the description below in the variable profileScale. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data.

Special values are defined as:
-9999.9 Missing value

**S1 (Swath)**

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value
Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCstatus (Group)
SCorientation (2-byte integer, array size: nscan):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

pixelStatus (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

0 : Valid pixel
1 : Invalid Latitude / Longitude
2 : Channel Tbs out of range
3 : Surface code / histogram mismatch
4 : Missing TCWV, T2m, or sfccode from preprocessor
5 : No Bayesian Solution
-99 : Missing value

qualityFlag (1-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 4).

Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution." Pixels can be set to 1 for the following reasons:
   - Sunglint is present, RFI, geolocate, warm load
     or other L1C 'positive value' quality warning flags.
   - All sea-ice covered surfaces.
   - All snow covered surfaces.
- Sensor channels are missing, but not critical ones.
2: "Use pixel with extreme care over snow covered surface."
   This is a special value for snow covered surfaces only.
The pixel is set to 2 if the probability of precipitation
is of poor quality or indeterminate. Use these pixels
for climatological averaging of precipitation, but not
for individual storm scale daily cases.
3: "Use with extreme caution." Pixels are set to 3 if
   they have channels missing critical to the retrieval,
   but the choice has been made to continue the retrieval
   for the pixel.
-99: Missing value

**L1CqualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.
0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
   Negative: Copied from negative 1C Quality flag (GMI only)

**surfaceTypeIndex** (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

   Codes include
   1: Ocean
   2: Sea-Ice
   3-7: Decreasing vegetation
   8-11: Decreasing snow cover
   12: Standing Water
   13: Land/ocean or water Coast
   14: Sea-ice edge
   -99: Missing value

**totalColumnWaterVaporIndex** (1-byte integer, array size: npixel x nscan):
The integer total precipitable water used to select the correct database profiles. Total-
ColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable
Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof
product the GANAL model is used. In the NRT Gprof product the JMAfcst model is
used. Values range from 0 to 78 mm. Special values are defined as:
-99: Missing value

**CAPE** (2-byte integer, array size: npixel x nscan):
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range
from 1 to 5. Special values are defined as:
   -9999  Missing value

**temp2mIndex** (2-byte integer, array size: npixel x nscan):
The 2 meter temperature Index used to select profiles in the database. Values are in K.
Special values are defined as:
   -9999  Missing value

**sunGlintAngle** (1-byte integer, array size: npixel x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. sunGlintAngle is the angular separation between the reflected satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values range from 0 to 127 degrees. Special values are defined as:
   -88  Sun below horizon
   -99  Missing

**probabilityOfPrecip** (1-byte integer, array size: npixel x nscan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
   -99  Missing value

**spare2** (2-byte integer, array size: npixel x nscan):
Spare variable.

**surfacePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval.
NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
   -9999.9  Missing value

**frozenPrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous frozen precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined from Combined Profiles addition of snow and grauple in the lowest profile level with a rate of fall factor. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
   -9999.9  Missing value

**convectivePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
   -9999.9  Missing value

**rainWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0
(not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

**cloudWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

**iceWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

**mostLikelyPrecipitation** (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip1stTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip2ndTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values are defined as:
-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:
-9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description below. Values range from 1 to 80. Special values are defined as:
-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel, select your species, level, and profile2mTempIndex,
then use profileNumber and profileScale
to obtain the value:

Where:
- S = species (1-5)
  Species defined in speciesDescription
- T = profile2mTempIndex (1-12)
  Temperatures defined in temperatureDescriptions
- L = profile level (1-28) Top of each level
  specified in hgtTopLayer
- P = profileNumber (1-80) for species S

In a Fortran program,
- P = profileNumber(S)
- Pixel Value = profileScale(S) * clusterProfiles(S,T,L,P)

In a C program,
- P = profileNumber[S-1]
- Pixel Value = profileScale[S] * clusterProfiles[P-1][L-1][T-1][S-1]

C Structure Header file:

```c
#ifndef _TK_2AGPROFAMSR2_H_
#define _TK_2AGPROFAMSR2_H_

#ifndef _SCSTATUS_
#define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
} SCANTIME;
#endif
```

```c
#endif
```
signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L2AGPROFAMSR2_S1_
#define _L2AGPROFAMSR2_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[486];
    float Longitude[486];
    SCSTATUS SCstatus;
    signed char pixelStatus[486];
    signed char qualityFlag[486];
    signed char L1CqualityFlag[486];
    signed char surfaceTypeIndex[486];
    signed char totalColumnWaterVaporIndex[486];
    short CAPE[486];
    short temp2mIndex[486];
    signed char sunGlintAngle[486];
    signed char probabilityOfPrecip[486];
    short spare2[486];
    float surfacePrecipitation[486];
    float frozenPrecipitation[486];
    float convectivePrecipitation[486];
    float rainWaterPath[486];
    float cloudWaterPath[486];
    float iceWaterPath[486];
    float mostLikelyPrecipitation[486];
    float precip1stTertial[486];
    float precip2ndTertial[486];
    short profileTemp2mIndex[486];
    short profileNumber[486][5];
    float profileScale[486][5];
} L2AGPROFAMSR2_S1;
#endif

 ifndef _GPROFDFHEADR_
 define _GPROFDFHEADR_

typedef struct {
    unsigned char speciesDescription[5][21];
    float hgtTopLayer[28];
    float temperatureDescriptions[12];
    float clusterProfiles[80][28][12][5];
} GPROFDFHEADR;

 #endif
 #endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude
    REAL*4 SCaltitude
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFAMSR2_S1/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(486)
    REAL*4 Longitude(486)
5.38  2AGPROFWIND - Radiometer Profiling

2AGPROFWIND, "Radiometer Profiling", generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2017. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 80 typical structures for each hydrometeor or heating profile. These vertical structures are referenced to as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written
for each pixel.

Two products use the 2AGPROFWIND format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Dimension definitions:

- nscan: var
  Number of scans in the granule.
- npixel: 80
  Number of pixels in each scan.
- nspecies: 5
  Number of hydrometeor species. Species are defined in speciesDescription in the DataHeader group.
- sddim: 21
  Number of characters in each species description.
- ntemps: 12
  Number of profile temperature indeces. Indeces are defined in temperatureDescriptions in the DataHeader group.
- nlyrs: 28
  Number of profiling layers. The top height of each layer is defined in hgtTopLayer in the DataHeader group.
- nprf: 80
  Number of unique profiles for each species and 2 meter Temperature index.

Figure through Figure 452 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):

FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):

InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):

NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):

FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**GprofInfo** (Metadata):

GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

**GprofDHeadr** (Group)
Figure 448: Data Format Structure for 2AGPROFWIND, Radiometer Profiling

continued on next figure

•

•
### continued from last figure

<table>
<thead>
<tr>
<th>Field</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>convectivePrecipitation</td>
<td>4 bytes</td>
<td>Array: npixel x nscan</td>
</tr>
<tr>
<td>rainWaterPath</td>
<td>4 bytes</td>
<td>Array: npixel x nscan</td>
</tr>
<tr>
<td>cloudWaterPath</td>
<td>4 bytes</td>
<td>Array: npixel x nscan</td>
</tr>
<tr>
<td>iceWaterPath</td>
<td>4 bytes</td>
<td>Array: npixel x nscan</td>
</tr>
<tr>
<td>mostLikelyPrecipitation</td>
<td>4 bytes</td>
<td>Array: npixel x nscan</td>
</tr>
<tr>
<td>precip1stTertiary</td>
<td>4 bytes</td>
<td>Array: npixel x nscan</td>
</tr>
<tr>
<td>precip2ndTertiary</td>
<td>4 bytes</td>
<td>Array: npixel x nscan</td>
</tr>
<tr>
<td>profileTemp2mIndex</td>
<td>2 bytes</td>
<td>Array: npixel x nscan</td>
</tr>
<tr>
<td>profileNumber</td>
<td>2 bytes</td>
<td>Array: nspecies x npixel x nscan</td>
</tr>
<tr>
<td>profileScale</td>
<td>4 bytes</td>
<td>Array: nspecies x npixel x nscan</td>
</tr>
</tbody>
</table>

Figure 449: Data Format Structure for 2AGPROFWIND, Radiometer Profiling

<table>
<thead>
<tr>
<th>Field</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>speciesDescription</td>
<td>1 byte</td>
<td>Array: sddim x nspecies</td>
</tr>
<tr>
<td>hgtTopLayer</td>
<td>4 bytes</td>
<td>Array: nlyrs</td>
</tr>
<tr>
<td>temperatureDescriptions</td>
<td>4 bytes</td>
<td>Array: ntemps</td>
</tr>
<tr>
<td>clusterProfiles</td>
<td>4 bytes</td>
<td>Array: nspecies x ntemps x nlyrs x nprf</td>
</tr>
</tbody>
</table>

Figure 450: Data Format Structure for 2AGPROFWIND, GprofDHeadr

<table>
<thead>
<tr>
<th>Field</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Year</td>
<td>2 bytes</td>
<td>Array: nscan</td>
</tr>
<tr>
<td>Month</td>
<td>1 byte</td>
<td>Array: nscan</td>
</tr>
<tr>
<td>DayOfMonth</td>
<td>1 byte</td>
<td>Array: nscan</td>
</tr>
<tr>
<td>Hour</td>
<td>1 byte</td>
<td>Array: nscan</td>
</tr>
<tr>
<td>Minute</td>
<td>1 byte</td>
<td>Array: nscan</td>
</tr>
<tr>
<td>Second</td>
<td>1 byte</td>
<td>Array: nscan</td>
</tr>
<tr>
<td>MilliSecond</td>
<td>2 bytes</td>
<td>Array: nscan</td>
</tr>
<tr>
<td>DayOfYear</td>
<td>2 bytes</td>
<td>Array: nscan</td>
</tr>
<tr>
<td>SecondOfDay</td>
<td>8 bytes</td>
<td>Array: nscan</td>
</tr>
</tbody>
</table>

Figure 451: Data Format Structure for 2AGPROFWIND, ScanTime
speciesDescription (1-byte char, array size: sddim x nspecies):
Description of each species. Special values are defined as:
   255    Missing value

hgtTopLayer (4-byte float, array size: nlyrs):
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are
every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ...
9.5, 10.0, 11.0, ... 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing
data. Values range from 0 to 18.0 km. Special values are defined as:
   -9999.9  Missing value

temperatureDescriptions (4-byte float, array size: ntemps):
Temperature of 2 meter temperature indeces of clusterProfiles. NOTE: In V05 -9999.0
(not -9999.9) was used to denote missing data. Values are in C. Special values are defined
as:
   -9999.9  Missing value

clusterProfiles (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2
meter temperature index (12); vertical layers (28); and profile number (80). To recover
values in a profile see the description below in the variable profileScale. NOTE: In V05
-9999.0 (not -9999.9) was used to denote missing data.

   Special values are defined as:
   -9999.9  Missing value

S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2
data products. See Metadata for GPM Products for details.
ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCstatus** (Group)

**SCorientation** (2-byte integer, array size: nscan):
The angle of the spacecraft vector \( (v) \) from the satellite forward direction of motion, measured clockwise facing down. The relationship of \( v \) to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

**SClatitude** (4-byte float, array size: nscan):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**SClongitude** (4-byte float, array size: nscan):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**SCaltitude** (4-byte float, array size: nscan):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, **FractionalGranuleNumber** = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**pixelStatus** (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, **pixelStatus** explains the reason (Range 0 - 99).

0 : Valid pixel
1 : Invalid Latitude / Longitude
2 : Channel Tbs out of range
3 : Surface code / histogram mismatch
4 : Missing TCWV, T2m, or sfccode from preprocessor
5 : No Bayesian Solution
-99 : Missing value

**qualityFlag** (1-byte integer, array size: npixel x nscan):
**qualityFlag** indicates a generalized quality of the retrieved pixel (Range 0 - 4).
Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution." Pixels can be set to 1 for the following reasons:
   - Sunglint is present, RFI, geolocate, warm load
   - or other L1C 'positive value' quality warning flags.
   - All sea-ice covered surfaces.
   - All snow covered surfaces.
   - Sensor channels are missing, but not critical ones.
2 : "Use pixel with extreme care over snow covered surface."
   This is a special value for snow covered surfaces only.
   The pixel is set to 2 if the probability of precipitation is of poor quality or indeterminate. Use these pixels for climatological averaging of precipitation, but not for individual storm scale daily cases.
3 : "Use with extreme caution." Pixels are set to 3 if they have channels missing critical to the retrieval, but the choice has been made to continue the retrieval for the pixel.
-99 : Missing value

**L1CqualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.

0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
Negative: Copied from negative 1C Quality flag (GMI only)

**surfaceTypeIndex** (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

   Codes include
   1 : Ocean
   2 : Sea-Ice
   3-7 : Decreasing vegetation
   8-11 : Decreasing snow cover
   12 : Standing Water
   13 : Land/ocean or water Coast
   14 : Sea-ice edge
   -99 : Missing value

**totalColumnWaterVaporIndex** (1-byte integer, array size: npixel x nscan):
The integer total precipitable water used to select the correct database profiles. Total-ColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable
Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof product the GANAL model is used. In the NRT Gprof product the JMAfcst model is used. Values range from 0 to 78 mm. Special values are defined as:
  -99   Missing value

**CAPE (2-byte integer, array size: npixel x nscan):**
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range from 1 to 5. Special values are defined as:
  -9999   Missing value

**temp2mIndex (2-byte integer, array size: npixel x nscan):**
The 2 meter temperature Index used to select profiles in the database. Values are in K. Special values are defined as:
  -9999   Missing value

**sunGlintAngle (1-byte integer, array size: npixel x nscan):**
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. sunGlintAngle is the angular separation between the reflected satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values range from 0 to 127 degrees. Special values are defined as:
  -88   Sun below horizon
  -99   Missing

**probabilityOfPrecip (1-byte integer, array size: npixel x nscan):**
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
  -99   Missing value

**spare2 (2-byte integer, array size: npixel x nscan):**
Spare variable.

**surfacePrecipitation (4-byte float, array size: npixel x nscan):**
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
  -9999.9   Missing value

**frozenPrecipitation (4-byte float, array size: npixel x nscan):**
The instantaneous frozen precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined from Combined Profiles addition of snow and graupel in the lowest profile level with a rate of fall factor. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
  -9999.9   Missing value

**convectivePrecipitation (4-byte float, array size: npixel x nscan):**
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a
valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**rainWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

**cloudWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

**iceWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

**mostLikelyPrecipitation** (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip1stTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip2ndTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values are defined as:
-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:
9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description
below. Values range from 1 to 80. Special values are defined as:
-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel, select your species, level, and profile2mTempIndex, then use profileNumber and profileScale to obtain the value:

Where:
- \( S = \) species (1-5)
  - Species defined in speciesDescription
- \( T = \) profile2mTempIndex (1-12)
  - Temperatures defined in temperatureDescriptions
- \( L = \) profile level (1-28) Top of each level specified in hgtTopLayer
- \( P = \) profileNumber (1-80) for species \( S \)

In a Fortran program,
\[
P = \text{profileNumber}(S)
\]
\[
\text{Pixel Value} = \text{profileScale}(S) \times \text{clusterProfiles}(S,T,L,P)
\]

In a C program,
\[
P = \text{profileNumber}[S-1]
\]
\[
\text{Pixel Value} = \text{profileScale}[S] \times \text{clusterProfiles}[P-1][L-1][T-1][S-1]
\]

**C Structure Header file:**

```c
#ifndef _TK_2AGPROFWIND_H_
define _TK_2AGPROFWIND_H_

#ifndef _SCSTATUS_
define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
```
#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L2AGPROFWIND_S1_
#define _L2AGPROFWIND_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[80];
    float Longitude[80];
    SCSTATUS SCstatus;
    signed char pixelStatus[80];
    signed char qualityFlag[80];
    signed char LiCqualityFlag[80];
    signed char surfaceTypeIndex[80];
    signed char totalColumnWaterVaporIndex[80];
    short CAPE[80];
    short temp2mIndex[80];
    signed char sunGlintAngle[80];
    signed char probabilityOfPrecip[80];
    short spare2[80];
    float surfacePrecipitation[80];
    float frozenPrecipitation[80];
    float convectivePrecipitation[80];
    float rainWaterPath[80];
    float cloudWaterPath[80];
    float iceWaterPath[80];

} L2AGPROFWIND_S1;

#endif
float mostLikelyPrecipitation[80];
float precip1stTertial[80];
float precip2ndTertial[80];
short profileTemp2mIndex[80];
short profileNumber[80][5];
float profileScale[80][5];
} L2AGPROFWIND_S1;

#endif

#ifndef _GPROFDHEADR_
#define _GPROFDHEADR_

typedef struct {
    unsigned char speciesDescription[5][21];
    float hgtTopLayer[28];
    float temperatureDescriptions[12];
    float clusterProfiles[80][28][12][5];
} GPROFDHEADR;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude
    REAL*4 SCaltitude
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear

REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFWIND_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(80)
  REAL*4 Longitude(80)
  RECORD /SCSTATUS/ SCstatus
  BYTE pixelStatus(80)
  BYTE qualityFlag(80)
  BYTE L1CqualityFlag(80)
  BYTE surfaceTypeIndex(80)
  BYTE totalColumnWaterVaporIndex(80)
  INTEGER*2 CAPE(80)
  INTEGER*2 temp2mIndex(80)
  BYTE sunGlintAngle(80)
  BYTE probabilityOfPrecip(80)
  INTEGER*2 spare2(80)
  REAL*4 surfacePrecipitation(80)
  REAL*4 frozenPrecipitation(80)
  REAL*4 convectivePrecipitation(80)
  REAL*4 rainWaterPath(80)
  REAL*4 cloudWaterPath(80)
  REAL*4 iceWaterPath(80)
  REAL*4 mostLikelyPrecipitation(80)
  REAL*4 precip1stTertial(80)
  REAL*4 precip2ndTertial(80)
  INTEGER*2 profileTemp2mIndex(80)
  INTEGER*2 profileNumber(5,80)
  REAL*4 profileScale(5,80)
END STRUCTURE

STRUCTURE /GPROFDHEADR/
  CHARACTER speciesDescription(21,5)
  REAL*4 hgtTopLayer(28)
  REAL*4 temperatureDescriptions(12)
  REAL*4 clusterProfiles(5,12,28,80)
END STRUCTURE
5.39 2AGPROFAMSUB - Radiometer Profiling

2AGPROFAMSUB, "Radiometer Profiling", generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2017. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 80 typical structures for each hydrometeor or heating profile. These vertical structures are referenced as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written for each pixel.

Two products use the 2AGPROFAMSUB format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Dimension definitions:

- `nscan`: Number of scans in the granule.
- `npixel`: Number of pixels in each scan.
- `nspecies`: Number of hydrometeor species. Species are defined in `speciesDescription` in the DataHeader group.
- `sddim`: Number of characters in each species description.
- `ntemps`: Number of profile temperature indeces. Indeces are defined in `temperatureDescriptions` in the DataHeader group.
- `nlyrs`: Number of profiling layers. The top height of each layer is defined in `hgtTopLayer` in the DataHeader group.
- `nprf`: Number of unique profiles for each species and 2 meter Temperature index.

Figure 453 through Figure 457 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
Figure 453: Data Format Structure for 2AGPROFAMSUB, Radiometer Profiling
continued from last figure

- convectivePrecipitation: 4 bytes, Array: npixel x nscan
- rainWaterPath: 4 bytes, Array: npixel x nscan
- cloudWaterPath: 4 bytes, Array: npixel x nscan
- iceWaterPath: 4 bytes, Array: npixel x nscan
- mostLikelyPrecipitation: 4 bytes, Array: npixel x nscan
- precip1stTertial: 4 bytes, Array: npixel x nscan
- precip2ndTertial: 4 bytes, Array: npixel x nscan
- profileTemp2mIndex: 2 bytes, Array: npixel x nscan
- profileNumber: 2 bytes, Array: nspecies x npixel x nscan
- profileScale: 4 bytes, Array: nspecies x npixel x nscan

Figure 454: Data Format Structure for 2AGPROFAMSUB, Radiometer Profiling

- speciesDescription: 1 byte, Array: sddim x nspecies
- hgtTopLayer: 4 bytes, Array: nlyrs
- temperatureDescriptions: 4 bytes, Array: ntemps
- clusterProfiles: 4 bytes, Array: nspecies x ntemps x nlyrs x nprf

Figure 455: Data Format Structure for 2AGPROFAMSUB, GprofDHeadr

- Year: 2 bytes, Array: nscan
- Month: 1 byte, Array: nscan
- DayOfMonth: 1 byte, Array: nscan
- Hour: 1 byte, Array: nscan
- Minute: 1 byte, Array: nscan
- Second: 1 byte, Array: nscan
- MilliSecond: 2 bytes, Array: nscan
- DayOfYear: 2 bytes, Array: nscan
- SecondOfDay: 8 bytes, Array: nscan

Figure 456: Data Format Structure for 2AGPROFAMSUB, ScanTime
**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**GprofInfo** (Metadata):
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

### GprofDHeadr (Group)

**speciesDescription** (1-byte char, array size: sddim x nspecies):
Description of each species. Special values are defined as:

- 255  Missing value

**hgtTopLayer** (4-byte float, array size: nlyrs):
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ... 9.5, 10.0, 11.0, ... 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 18.0 km. Special values are defined as:

- -9999.9  Missing value

**temperatureDescriptions** (4-byte float, array size: ntemps):
Temperature of 2 meter temperature indeces of clusterProfiles. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in C. Special values are defined as:

- -9999.9  Missing value

**clusterProfiles** (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2 meter temperature index (12); vertical layers (28); and profile number (80). To recover values in a profile see the description below in the variable profileScale. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data.

Special values are defined as:

- -9999.9  Missing value
S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value
Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group)

SCorientation (2-byte integer, array size: nscan):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

pixelStatus (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

0 : Valid pixel
1 : Invalid Latitude / Longitude
2 : Channel Tbs out of range
3 : Surface code / histogram mismatch
4 : Missing TCWV, T2m, or sfccode from preprocessor
5 : No Bayesian Solution
-99 : Missing value

**qualityFlag** (1-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 4).

Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution." Pixels can be set to 1 for the following reasons:
   - Sunglint is present, RFI, geolocate, warm load
   - or other L1C 'positive value' quality warning flags.
   - All sea-ice covered surfaces.
   - All snow covered surfaces.
   - Sensor channels are missing, but not critical ones.
2 : "Use pixel with extreme care over snow covered surface."
   This is a special value for snow covered surfaces only.
   The pixel is set to 2 if the probability of precipitation
   is of poor quality or indeterminate. Use these pixels
   for climatological averaging of precipitation, but not
   for individual storm scale daily cases.
3 : "Use with extreme caution." Pixels are set to 3 if
   they have channels missing critical to the retrieval,
   but the choice has been made to continue the retrieval
   for the pixel.
-99 : Missing value

**L1CQualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.

0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
Negative: Copied from negative 1C Quality flag (GMI only)

**surfaceTypeIndex** (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

Codes include
1 : Ocean
2 : Sea-Ice
3-7 : Decreasing vegetation
8-11 : Decreasing snow cover
12 : Standing Water
13 : Land/ocean or water Coast
14 : Sea-ice edge
-99 : Missing value

**totalColumnWaterVaporIndex** (1-byte integer, array size: npixel x nscan):
The integer total precipitable water used to select the correct database profiles. Total-ColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof product the GANAL model is used. In the NRT Gprof product the JMAfcst model is used. Values range from 0 to 78 mm. Special values are defined as:
-99 Missing value

**CAPE** (2-byte integer, array size: npixel x nscan):
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range from 1 to 5. Special values are defined as:
-9999 Missing value

**temp2mIndex** (2-byte integer, array size: npixel x nscan):
The 2 meter temperature Index used to select profiles in the database. Values are in K. Special values are defined as:
-9999 Missing value

**sunGlintAngle** (1-byte integer, array size: npixel x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. sunGlintAngle is the angular separation between the reflected satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values range from 0 to 127 degrees. Special values are defined as:
-88 Sun below horizon
-99 Missing

**probabilityOfPrecip** (1-byte integer, array size: npixel x nscan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
-99 Missing value

**spare2** (2-byte integer, array size: npixel x nscan):
Spare variable.

**surfacePrecipitation** (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value
frozenPrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous frozen precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined from Combined Profiles addition of snow and graupel in the lowest profile level with a rate of fall factor. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

convectivePrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

rainWaterPath (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

cloudWaterPath (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

iceWaterPath (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

mostLikelyPrecipitation (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precip1stTertial (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precip2ndTertial (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values
are defined as:
-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:
-9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description below. Values range from 1 to 80. Special values are defined as:
-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel, select your species, level, and profile2mTempIndex, then use profileNumber and profileScale to obtain the value:

Where:
\[
S = \text{species (1-5)}
\]
Species defined in speciesDescription
\[
T = \text{profile2mTempIndex (1-12)}
\]
Temperatures defined in temperatureDescriptions
\[
L = \text{profile level (1-28) Top of each level specified in hgtTopLayer}
\]
\[
P = \text{profileNumber (1-80) for species S}
\]

In a Fortran program,
\[
P = \text{profileNumber}(S)
\]
Pixel Value = profileScale(S) * clusterProfiles(S,T,L,P)

In a C program,
\[
P = \text{profileNumber}[S-1]
\]
Pixel Value = profileScale[S] * clusterProfiles[P-1][L-1][T-1][S-1]

**C Structure Header file:**

```c
#ifndef _TK_2AGPROFAMSUB_H_
#define _TK_2AGPROFAMSUB_H_

#endif
```

```c
#ifndef _SCSTATUS_
#define _SCSTATUS_

#endif
```

```c
```
typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L2AGPROFAMSUB_S1_
#define _L2AGPROFAMSUB_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[90];
    float Longitude[90];
    SCSTATUS SCstatus;
    signed char pixelStatus[90];
    signed char qualityFlag[90];
    signed char L1CqualityFlag[90];
    signed char surfaceTypeIndex[90];
    signed char totalColumnWaterVaporIndex[90];
    short CAPE[90];
    short temp2mIndex[90];
}
signed char sunGlintAngle[90];
signed char probabilityOfPrecip[90];
short spare2[90];
float surfacePrecipitation[90];
float frozenPrecipitation[90];
float convectivePrecipitation[90];
float rainWaterPath[90];
float cloudWaterPath[90];
float iceWaterPath[90];
float mostLikelyPrecipitation[90];
float precip1stTertial[90];
float precip2ndTertial[90];
short profileTemp2mIndex[90];
short profileNumber[90][5];
float profileScale[90][5];
} L2AGPROFAMSU3 S1;

#endif

#ifndef _GPROFDHEADR_
#define _GPROFDHEADR_

typedef struct {
    unsigned char speciesDescription[5][21];
    float hgtTopLayer[28];
    float temperatureDescriptions[12];
    float clusterProfiles[80][28][12][5];
} GPROFDHEADR;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /SCSTATUS/
    INTEGER*2 SCorientation
    REAL*4 SClatitude
    REAL*4 SClongitude
    REAL*4 SCaltitude
    REAL*8 FractionalGranuleNumber
END STRUCTURE
STRUCTURE /SCANTIME/
   INTEGER*2 Year
   BYTE Month
   BYTE DayOfMonth
   BYTE Hour
   BYTE Minute
   BYTE Second
   INTEGER*2 MilliSecond
   INTEGER*2 DayOfYear
   REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFAMSUB_S1/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(90)
   REAL*4 Longitude(90)
   RECORD /SCSTATUS/ SCstatus
   BYTE pixelStatus(90)
   BYTE qualityFlag(90)
   BYTE L1CqualityFlag(90)
   BYTE surfaceTypeIndex(90)
   BYTE totalColumnWaterVaporIndex(90)
   INTEGER*2 CAPE(90)
   INTEGER*2 temp2mIndex(90)
   BYTE sunGlintAngle(90)
   BYTE probabilityOfPrecip(90)
   INTEGER*2 spare2(90)
   REAL*4 surfacePrecipitation(90)
   REAL*4 frozenPrecipitation(90)
   REAL*4 convectivePrecipitation(90)
   REAL*4 surfacePrecipitation(90)
   REAL*4 rainWaterPath(90)
   REAL*4 cloudWaterPath(90)
   REAL*4 iceWaterPath(90)
   REAL*4 mostLikelyPrecipitation(90)
   REAL*4 precip1stTertial(90)
   REAL*4 precip2ndTertial(90)
   INTEGER*2 profileTemp2mIndex(90)
   INTEGER*2 profileNumber(5,90)
   REAL*4 profileScale(5,90)
END STRUCTURE

STRUCTURE /GPROFDHEADR/
   CHARACTER speciesDescription(21,5)
REAL*4 hgtTopLayer(28)
REAL*4 temperatureDescriptions(12)
REAL*4 clusterProfiles(5,12,28,80)
END STRUCTURE

5.40 2AGPROFATMS - Radiometer Profiling

2AGPROFATMS, ”Radiometer Profiling”, generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2014. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 100 typical structures for each hydrometeor or heating profile. These vertical structures are referenced to as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written for each pixel.

Two products use this format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Note that the 3 outer pixels on each side of the swath are set to missing. I.e., 6 pixels in each swath are set to missing.

**Dimension definitions:**

- nscan var Number of scans in the granule.
- npixel 96 Number of pixels in each scan.
- nspecies 5 Number of hydrometeor species. Species are defined in speciesDescription in the DataHeader group.
- sddim 21 Number of characters in each species description.
- ntemps 12 Number of profile temperature indeces. Indeces are defined in temperatureDescriptions in the DataHeader group.
- nlyrs 28 Number of profiling layers. The top height of each layer is defined in hgtTopLayer in the DataHeader group.
- nprf 80 Number of unique profiles for each species and 2 meter Temperature index.

Figure 458 through Figure 462 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**

FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.
Figure 458: Data Format Structure for 2AGPROFATMS, Radiometer Profiling
continued from last figure

- convectivePrecipitation: 4 bytes, Array: npixel x nscan
- rainWaterPath: 4 bytes, Array: npixel x nscan
- cloudWaterPath: 4 bytes, Array: npixel x nscan
- iceWaterPath: 4 bytes, Array: npixel x nscan
- mostLikelyPrecipitation: 4 bytes, Array: npixel x nscan
- precip1stTertial: 4 bytes, Array: npixel x nscan
- precip2ndTertial: 4 bytes, Array: npixel x nscan
- profileTemp2mIndex: 2 bytes, Array: npixel x nscan
- profileNumber: 2 bytes, Array: nspecies x npixel x nscan
- profileScale: 4 bytes, Array: nspecies x npixel x nscan

Figure 459: Data Format Structure for 2AGPROFATMS, Radiometer Profiling

- speciesDescription: 1 byte, Array: sddim x nspecies
- hgtTopLayer: 4 bytes, Array: nlyrs
- temperatureDescriptions: 4 bytes, Array: ntemps
- clusterProfiles: 4 bytes, Array: nspecies x ntemps x nlyrs x nprf

Figure 460: Data Format Structure for 2AGPROFATMS, GprofDHeadr

- Year: 2 bytes, Array: nscan
- Month: 1 byte, Array: nscan
- DayOfMonth: 1 byte, Array: nscan
- Hour: 1 byte, Array: nscan
- Minute: 1 byte, Array: nscan
- Second: 1 byte, Array: nscan
- MilliSecond: 2 bytes, Array: nscan
- DayOfYear: 2 bytes, Array: nscan
- SecondOfDay: 8 bytes, Array: nscan

Figure 461: Data Format Structure for 2AGPROFATMS, ScanTime
**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**GprofInfo** (Metadata):
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

---

**GprofDHeadr** (Group)

**speciesDescription** (1-byte char, array size: sddim x nspecies):
Description of each species. Special values are defined as:
- 255 Missing value

**hgtTopLayer** (4-byte float, array size: nlyrs):
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ..., 9.5, 10.0, 11.0, ..., 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 18.0 km. Special values are defined as:
- -9999.9 Missing value

**temperatureDescriptions** (4-byte float, array size: ntemps):
Temperature of 2 meter temperature indeces of clusterProfiles. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in C. Special values are defined
as:
-9999.9 Missing value

clusterProfiles (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2
meter temperature index (12); vertical layers (28); and profile number (80). To recover
values in a profile see the description below in the variable profileScale. NOTE: In V05
-9999.0 (not -9999.9) was used to denote missing data.

Special values are defined as:
-9999.9 Missing value

S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2
data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined
as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value
MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group)

SCorientation (2-byte integer, array size: nscan):
The angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. The relationship of v to the sensor geometry is defined in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values are defined as:
-9999 Missing value

SClatitude (4-byte float, array size: nscan):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

SClongitude (4-byte float, array size: nscan):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCaltitude (4-byte float, array size: nscan):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9 Missing value
**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft's trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**pixelStatus** (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

0 : Valid pixel
1 : Invalid Latitude / Longitude
2 : Channel Tbs out of range
3 : Surface code / histogram mismatch
4 : Missing TCWV, T2m, or sfccode from preprocessor
5 : No Bayesian Solution
-99 : Missing value

**qualityFlag** (1-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 4).

Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution." Pixels can be set to 1 for the following reasons:
   - Sunglint is present, RFI, geolocate, warm load
   or other L1C 'positive value' quality warning flags.
   - All sea-ice covered surfaces.
   - All snow covered surfaces.
   - Sensor channels are missing, but not critical ones.
2 : "Use pixel with extreme care over snow covered surface."
   This is a special value for snow covered surfaces only.
The pixel is set to 2 if the probability of precipitation
is of poor quality or indeterminate. Use these pixels
for climatological averaging of precipitation, but not
for individual storm scale daily cases.
3 : "Use with extreme caution." Pixels are set to 3 if
   they have channels missing critical to the retrieval,
   but the choice has been made to continue the retrieval
   for the pixel.
-99 : Missing value

**L1CqualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.
0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
Negative: Copied from negative 1C Quality flag (GMI only)

**surfaceTypeIndex** (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

Codes include
1 : Ocean
2 : Sea-Ice
3–7 : Decreasing vegetation
8–11 : Decreasing snow cover
12 : Standing Water
13 : Land/ocean or water Coast
14 : Sea-ice edge
-99 : Missing value

**totalColumnWaterVaporIndex** (1-byte integer, array size: npixel x nscan):
The integer total precipitable water used to select the correct database profiles. Total-ColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof product the GANAL model is used. In the NRT Gprof product the JMAfcst model is used. Values range from 0 to 78 mm. Special values are defined as:
-99 Missing value

**CAPE** (2-byte integer, array size: npixel x nscan):
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range from 1 to 5. Special values are defined as:
-9999 Missing value

**temp2mIndex** (2-byte integer, array size: npixel x nscan):
The 2 meter temperature Index used to select profiles in the database. Values are in K. Special values are defined as:
-9999 Missing value

**sunGlintAngle** (1-byte integer, array size: npixel x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected off the Earth’s surface. sunGlintAngle is the angular separation between the reflected satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values range from 0 to 127 degrees. Special values are defined as:
-88 Sun below horizon
-99 Missing
probabilityOfPrecip (1-byte integer, array size: npixel x nscan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
-99 Missing value

spare2 (2-byte integer, array size: npixel x nscan):
Spare variable.

surfacePrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

totalPrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous total precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

convectivePrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

rainWaterPath (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

cloudWaterPath (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

iceWaterPath (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

mostLikelyPrecipitation (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval.
NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip1stTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip2ndTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values are defined as:
-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:
-9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description below. Values range from 1 to 80. Special values are defined as:
-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel, select your species, level, and profile2mTempIndex, then use profileNumber and profileScale to obtain the value:

Where:
- \( S = \) species \((1-5)\)
  - Species defined in speciesDescription
- \( T = \) profile2mTempIndex \((1-12)\)
  - Temperatures defined in temperatureDescriptions
- \( L = \) profile level \((1-28)\) Top of each level specified in hgtTopLayer
- \( P = \) profileNumber \((1-80)\) for species \( S \)

In a Fortran program,
- \( P = \) profileNumber\((S)\)
- Pixel Value = profileScale\((S) \times \text{clusterProfiles}(S,T,L,P)\)
In a C program,
\[ P = \text{profileNumber}[S-1] \]
\[ \text{Pixel Value} = \text{profileScale}[S] \times \text{clusterProfiles}[P-1][L-1][T-1][S-1] \]

**C Structure Header file:**

```c
#ifndef _TK_2AGPROFATMS_H_
define _TK_2AGPROFATMS_H_

#ifndef _SCSTATUS_
define _SCSTATUS_

typedef struct {
    short SCorientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
#endif
#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif
#endif

#ifndef _L2AGPROFATMS_S1_
define _L2AGPROFATMS_S1_

typedef struct {
    
```
SCANTIME ScanTime;
float Latitude[96];
float Longitude[96];
SCSTATUS SCstatus;
signed char pixelStatus[96];
signed char qualityFlag[96];
signed char L1CqualityFlag[96];
signed char surfaceTypeIndex[96];
signed char totalColumnWaterVaporIndex[96];
short CAPE[96];
short temp2mIndex[96];
signed char sunGlintAngle[96];
signed char probabilityOfPrecip[96];
short spare2[96];
float surfacePrecipitation[96];
float frozenPrecipitation[96];
float convectivePrecipitation[96];
float rainWaterPath[96];
float cloudWaterPath[96];
float iceWaterPath[96];
float mostLikelyPrecipitation[96];
float precip1stTertial[96];
float precip2ndTertial[96];
short profileTemp2mIndex[96];
short profileNumber[96][5];
float profileScale[96][5];
} L2AGPROFATMS_S1;

#endif

#ifndef _GPROFDHEADR_
#define _GPROFDHEADR_

typedef struct {
  unsigned char speciesDescription[5][21];
  float hgtTopLayer[28];
  float temperatureDescriptions[12];
  float clusterProfiles[80][28][12][5];
} GPROFDHEADR;
#endif

#endif
Fortran Structure Header file:

STRUCTURE /SCSTATUS/
  INTEGER*2 SCorientation
  REAL*4 SClatitude
  REAL*4 SClongitude
  REAL*4 SCaltitude
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFATMS_S1/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(96)
  REAL*4 Longitude(96)
  RECORD /SCSTATUS/ SCstatus
  BYTE pixelStatus(96)
  BYTE qualityFlag(96)
  BYTE L1CqualityFlag(96)
  BYTE surfaceTypeIndex(96)
  INTEGER*2 CAPE(96)
  INTEGER*2 temp2mIndex(96)
  BYTE sunGlintAngle(96)
  BYTE probabilityOfPrecip(96)
  INTEGER*2 spare2(96)
  REAL*4 surfacePrecipitation(96)
  REAL*4 frozenPrecipitation(96)
  REAL*4 convectivePrecipitation(96)
  REAL*4 rainWaterPath(96)
  REAL*4 cloudWaterPath(96)
  REAL*4 iceWaterPath(96)
REAL*4 mostLikelyPrecipitation(96)
REAL*4 precip1stTertial(96)
REAL*4 precip2ndTertial(96)
INTEGER*2 profileTemp2mIndex(96)
INTEGER*2 profileNumber(5,96)
REAL*4 profileScale(5,96)
END STRUCTURE

STRUCTURE /GPROFDHEADR/
    CHARACTER speciesDescription(21,5)
    REAL*4 hgtTopLayer(28)
    REAL*4 temperatureDescriptions(12)
    REAL*4 clusterProfiles(5,12,28,80)
END STRUCTURE

5.41 2AGPROFMHS - Radiometer Profiling

2AGPROFMHS, ”Radiometer Profiling”, generates surface rainfall and vertical hydrometeor profiles on a pixel by pixel basis from radiometer brightness temperature data using the Goddard Profiling algorithm GPROF2014. Because the vertical information comes from a radiometer, it is not written out in independent vertical layers like the TRMM Precipitation Radar. Instead, the output is referenced to one of 100 typical structures for each hydrometeor or heating profile. These vertical structures are referenced to as profiles in the output structure. Vertical hydrometeor profiles can be reconstructed to 28 layers by knowing the profile number (i.e. shape) of the profile and a scale factor that is written for each pixel.

Two products use this format: the regular product and the climate product. The regular product’s filename starts with 2A and its input includes GANAL data. The climate product’s filename starts with 2A-CLIM and its input includes ECMWF data.

Dimension definitions:

- nscan var Number of scans in the granule.
- npixel 90 Number of pixels in each scan.
- nspecies 5 Number of hydrometeor species. Species are defined in speciesDescription in the DataHeader group.
- sddim 21 Number of characters in each species description.
- ntemps 12 Number of profile temperature indeces. Indeces are defined in temperatureDescriptions in the DataHeader group.
- nlyrs 28 Number of profiling layers. The top height of each layer is defined in hgtTopLayer in the DataHeader group.
- nprf 80 Number of unique profiles for each species and 2 meter Temperature index.
Figure 463 through Figure 467 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**GprofInfo (Metadata):**
GprofInfo contains metadata required by Gprof. Used by 2A12 only. See Metadata for GPM Products for details.

**GprofDHeadr (Group)**

**speciesDescription (1-byte char, array size: sddim x nspecies):**
Description of each species. Special values are defined as:
- 255 Missing value

**hgtTopLayer (4-byte float, array size: nlyrs):**
Height of the top of each of 28 atmospheric layers in the clusterProfiles. The tops are every 0.5 km up to 10 km, then every km after that up to 18.0 km. Values are: 0.5, 1.0, ... 9.5, 10.0, 11.0, ... 18.0. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 18.0 km. Special values are defined as:
- -9999.9 Missing value

**temperatureDescriptions (4-byte float, array size: ntemps):**
Temperature of 2 meter temperature indeces of clusterProfiles. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in C. Special values are defined as:
- -9999.9 Missing value

**clusterProfiles (4-byte float, array size: nspecies x ntemps x nlyrs x nprf):**
Standard GPM profile structures. Dimensions are hydrometeor/heating species (5); 2
Figure 463: Data Format Structure for 2AGPROFMHS, Radiometer Profiling
continued from last figure

- convectivePrecipitation: 4 bytes, Array: npixel x nscan
- rainWaterPath: 4 bytes, Array: npixel x nscan
- cloudWaterPath: 4 bytes, Array: npixel x nscan
- iceWaterPath: 4 bytes, Array: npixel x nscan
- mostLikelyPrecipitation: 4 bytes, Array: npixel x nscan
- precip1stTertial: 4 bytes, Array: npixel x nscan
- precip2ndTertial: 4 bytes, Array: npixel x nscan
- profileTemp2mIndex: 2 bytes, Array: npixel x nscan
- profileNumber: 2 bytes, Array: nspecies x npixel x nscan
- profileScale: 4 bytes, Array: nspecies x npixel x nscan

Figure 464: Data Format Structure for 2AGPROFMHS, Radiometer Profiling

- speciesDescription: 1 byte, Array: sddim x nspecies
- hgtTopLayer: 4 bytes, Array: nlyrs
- temperatureDescriptions: 4 bytes, Array: ntemps
- clusterProfiles: 4 bytes, Array: nspecies x ntemps x nlyrs x nprf

Figure 465: Data Format Structure for 2AGPROFMHS, GprofDHeadr

- Year: 2 bytes, Array: nscan
- Month: 1 byte, Array: nscan
- DayOfMonth: 1 byte, Array: nscan
- Hour: 1 byte, Array: nscan
- Minute: 1 byte, Array: nscan
- Second: 1 byte, Array: nscan
- MilliSecond: 2 bytes, Array: nscan
- DayOfYear: 2 bytes, Array: nscan
- SecondOfDay: 8 bytes, Array: nscan

Figure 466: Data Format Structure for 2AGPROFMHS, ScanTime
meter temperature index (12); vertical layers (28); and profile number (80). To recover
values in a profile see the description below in the variable profileScale. NOTE: In V05
-9999.0 (not -9999.9) was used to denote missing data.

Special values are defined as:
-9999.9  Missing value

S1 (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2
data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

SCstatus (Group)

SCorientation (2-byte integer, array size: nscan):
The angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. The relationship of v to the sensor geometry is defined
in the introduction to this algorithm. Values range from 0 to 360 degrees. Special values
are defined as:
-9999 Missing value

SCLatitude (4-byte float, array size: nscan):
Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value
SClongitude (4-byte float, array size: nscan):
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

SCaltitude (4-byte float, array size: nscan):
Values range from 0 to 1000 km. Special values are defined as:
-9999.9  Missing value

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the
spacecraft is halfway through granule 10 and starting the descending half of the granule.
Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

pixelStatus (1-byte integer, array size: npixel x nscan):
If there is no retrieval at a given pixel, pixelStatus explains the reason (Range 0 - 99).

  0 : Valid pixel
  1 : Invalid Latitude / Longitude
  2 : Channel Tbs out of range
  3 : Surface code / histogram mismatch
  4 : Missing TCWV, T2m, or sfccode from preprocessor
  5 : No Bayesian Solution
-99 : Missing value

qualityFlag (1-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 4).

Valid values include:
0 : Pixel is "good" and has the highest confidence of the best retrieval.
1 : "Use with caution."  Pixels can be set to 1 for the following reasons:
- Sunglint is present, RFI, geolocate, warm load
  or other L1C 'positive value' quality warning flags.
- All sea-ice covered surfaces.
- All snow covered surfaces.
- Sensor channels are missing, but not critical ones.
2 : "Use pixel with extreme care over snow covered surface."
This is a special value for snow covered surfaces only.
The pixel is set to 2 if the probability of precipitation
is of poor quality or indeterminate. Use these pixels
for climatological averaging of precipitation, but not
for individual storm scale daily cases.
3 : "Use with extreme caution."  Pixels are set to 3 if
they have channels missing critical to the retrieval,
but the choice has been made to continue the retrieval
for the pixel.
-99 : Missing value

**L1CqualityFlag** (1-byte integer, array size: npixel x nscan):
Based on the pixel quality from the input L1C data file. Range is -128 to 127.

0: Normal
1: Positive 1C Quality flag
3: Negative 1C Quality flag (not GMI)
Negative: Copied from negative 1C Quality flag (GMI only)

**surfaceTypeIndex** (1-byte integer, array size: npixel x nscan):
Indicates the type of surface (Range 0 - 99).

Codes include
1 : Ocean
2 : Sea-Ice
3-7 : Decreasing vegetation
8-11 : Decreasing snow cover
12 : Standing Water
13 : Land/ocean or water Coast
14 : Sea-ice edge
-99 : Missing value

**totalColumnWaterVaporIndex** (1-byte integer, array size: npixel x nscan):
The integer total precipitable water used to select the correct database profiles. Total-
ColumnWaterVaporIndex is the nearest integer value to the model Total Precipitable
Water. In the climate Gprof product the ECMWF model is used. In the standard Gprof
product the GANAL model is used. In the NRT Gprof product the JMAfcst model is
used. Values range from 0 to 78 mm. Special values are defined as:
-99 Missing value

**CAPE** (2-byte integer, array size: npixel x nscan):
Model derived CAPE index. NOTE: In V05 CAPE is set to all missing. Values range
from 1 to 5. Special values are defined as:
-9999 Missing value

**temp2mIndex** (2-byte integer, array size: npixel x nscan):
The 2 meter temperature Index used to select profiles in the database. Values are in K.
Special values are defined as:
-9999 Missing value

**sunGlintAngle** (1-byte integer, array size: npixel x nscan):
Conceptually, the angle between the sun and the instrument view direction as reflected
off the Earth’s surface. sunGlintAngle is the angular separation between the reflected
satellite view vector and the sun vector. When sunGlintAngle is zero, the instrument views the center of the specular (mirror-like) sun reflection. If this angle is less than ten degrees, the pixel is affected by sunglint and the pixel’s qualityFlag is lowered to 1. Values range from 0 to 127 degrees. Special values are defined as:
- 88 Sun below horizon
- 99 Missing

probabilityOfPrecip (1-byte integer, array size: npixel x nscan):
A diagnostic variable, in percent, defining the fraction of raining vs. non-raining Database profiles that make up the final solution. Values range from 0 to 100 percent. Special values are defined as:
- 99 Missing value

spare2 (2-byte integer, array size: npixel x nscan):
Spare variable.

surfacePrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Check pixelStatus for a valid retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
- 9999.9 Missing value

frozenPrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous frozen precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined from Combined Profiles addition of snow and graupel in the lowest profile level with a rate of fall factor. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
- 9999.9 Missing value

convectivePrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous convective precipitation rate at the surface. Check pixelStatus for a valid retrieval. Defined using Combined/DPR precipitation type. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
- 9999.9 Missing value

rainWaterPath (4-byte float, array size: npixel x nscan):
Total integrated rain water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m^2. Special values are defined as:
- 9999.9 Missing value

cloudWaterPath (4-byte float, array size: npixel x nscan):
Total integrated cloud liquid water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m^2. Special values are defined as:
- 9999.9 Missing value
5.41 2AGPROFMHS - Radiometer Profiling

**iceWaterPath** (4-byte float, array size: npixel x nscan):
Total integrated ice water in the vertical atmospheric column. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values range from 0 to 3000 kg/m². Special values are defined as:
-9999.9 Missing value

**mostLikelyPrecipitation** (4-byte float, array size: npixel x nscan):
The surface precipitation value with the closest Tb match within the Bayesian retrieval. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip1stTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 1st tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precip2ndTertial** (4-byte float, array size: npixel x nscan):
The surface precipitation value at the 2nd tertiary of the precipitation distribution. NOTE: In V05 -9999.0 (not -9999.9) was used to denote missing data. Special values are defined as:
-9999.9 Missing value

**profileTemp2mIndex** (2-byte integer, array size: npixel x nscan):
Temperature 2 meter height Index in the clusterProfiles array. See profileScale description below. Values range from 1 to 21. Special values are defined as:
-9999 Missing value

**profileNumber** (2-byte integer, array size: nspecies x npixel x nscan):
Profile Number in the clusterProfiles array for each species. See profileScale description below. Values range from 1 to 80. Special values are defined as:
-9999 Missing value

**profileScale** (4-byte float, array size: nspecies x npixel x nscan):
profileScale is used to scale the values of the clusterProfiles array.

In order to recover a value of a single pixel, select your species, level, and profile2mTempIndex, then use profileNumber and profileScale to obtain the value:

Where:

- **S** = species (1-5)
  - Species defined in speciesDescription
- **T** = profile2mTempIndex (1-12)
  - Temperatures defined in temperatureDescriptions
- **L** = profile level (1-28) Top of each level
specified in hgtTopLayer
P = profileNumber (1-80) for species S

In a Fortran program,
P = profileNumber(S)
Pixel Value = profileScale(S) * clusterProfiles(S,T,L,P)

In a C program,
P = profileNumber[S-1]
Pixel Value = profileScale[S] * clusterProfiles[P-1][L-1][T-1][S-1]

C Structure Header file:

```c
#ifndef _TK_2AGPROFMHS_H_
define _TK_2AGPROFMHS_H_

#ifndef _SCSTATUS_
define _SCSTATUS_

typedef struct {
    short SCorrientation;
    float SClatitude;
    float SClongitude;
    float SCaltitude;
    double FractionalGranuleNumber;
} SCSTATUS;
#endif

#undef _SCSTATUS_
#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif
```

#ifndef _L2AGPROFMHS_S1_
#define _L2AGPROFMHS_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[90];
    float Longitude[90];
    SCSTATUS SCstatus;
    signed char pixelStatus[90];
    signed char qualityFlag[90];
    signed char L1CqualityFlag[90];
    signed char surfaceTypeIndex[90];
    signed char totalColumnWaterVaporIndex[90];
    short CAPE[90];
    short temp2mIndex[90];
    signed char sunGlintAngle[90];
    signed char probabilityOfPrecip[90];
    short spare2[90];
    float surfacePrecipitation[90];
    float frozenPrecipitation[90];
    float convectivePrecipitation[90];
    float rainWaterPath[90];
    float cloudWaterPath[90];
    float iceWaterPath[90];
    float mostLikelyPrecipitation[90];
    float precip1stTertial[90];
    float precip2ndTertial[90];
    short profileTemp2mIndex[90];
    short profileNumber[90][5];
    float profileScale[90][5];
} L2AGPROFMHS_S1;
#endif

#ifndef _GPROFDHEADR_
#define _GPROFDHEADR_

typedef struct {
    unsigned char speciesDescription[5][21];
    float hgtTopLayer[28];
} _GPROFDHEADR_
#endif
float temperatureDescriptions[12];
float clusterProfiles[80][28][12][5];
}
} GPROFHEADER;
#endif
#endif

Fortran Structure Header file:

```
STRUCTURE /SCSTATUS/
   INTEGER*2 SCorientation
   REAL*4 SClatitude
   REAL*4 SClongitude
   REAL*4 SCaltitude
   REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
   INTEGER*2 Year
   BYTE Month
   BYTE DayOfMonth
   BYTE Hour
   BYTE Minute
   BYTE Second
   BYTE MilliSecond
   INTEGER*2 DayOfYear
   REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AGPROFMHS_S1/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(90)
   REAL*4 Longitude(90)
   RECORD /SCSTATUS/ SCstatus
   BYTE pixelStatus(90)
   BYTE qualityFlag(90)
   BYTE L1CqualityFlag(90)
   BYTE surfaceTypeIndex(90)
   INTEGER*2 CAPE(90)
   INTEGER*2 temp2mIndex(90)
   BYTE sunGlintAngle(90)
```
BYTE probabilityOfPrecip(90)
INTEGER*2 spare2(90)
REAL*4 surfacePrecipitation(90)
REAL*4 frozenPrecipitation(90)
REAL*4 convectivePrecipitation(90)
REAL*4 rainWaterPath(90)
REAL*4 cloudWaterPath(90)
REAL*4 iceWaterPath(90)
REAL*4 mostLikelyPrecipitation(90)
REAL*4 precip1stTertial(90)
REAL*4 precip2ndTertial(90)
INTEGER*2 profileTemp2mIndex(90)
INTEGER*2 profileNumber(5,90)
REAL*4 profileScale(5,90)
END STRUCTURE

STRUCTURE /GPROFDHEADR/
  CHARACTER speciesDescription(21,5)
  REAL*4 hgtTopLayer(28)
  REAL*4 temperatureDescriptions(12)
  REAL*4 clusterProfiles(5,12,28,80)
END STRUCTURE

5.42 2APRPSSAPHIR - Radiometer Profiling

2APRPSSAPHIR, ”Radiometer Profiling”, generates surface rainfall on a pixel by pixel basis from radiometer brightness temperature data using the Precipitation Retrieval Profile Scheme PRPS2017.

Dimension definitions:
nscan var Number of scans in the granule.
npixel 182 Number of pixels in each scan.

Figure 468 through Figure 469 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information
Figure 468: Data Format Structure for 2APRPSSAPHIR, Radiometer Profiling

Figure 469: Data Format Structure for 2APRPSSAPHIR, ScanTime
separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**PRPSinfo** (Metadata):
PRPSinfo contains metadata required by PRPS. Used by 2APRPS products. See Metadata for GPM Products for details.

**S1** (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value
Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
   -99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
   -9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
   -9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
   -9999.9 Missing value

Latitude (4-byte float, array size: npixel x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
   -9999.9 Missing value

Longitude (4-byte float, array size: npixel x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9 Missing value

qualityFlag (2-byte integer, array size: npixel x nscan):
qualityFlag indicates a generalized quality of the retrieved pixel (Range 0 - 2).

Valid values include:
0 : All is OK
1 : Bad Tcs
2 : Altitude too high
-999 : Missing value

surfacePrecipitation (4-byte float, array size: npixel x nscan):
The instantaneous precipitation rate at the surface. Values are in mm/hr. Special values are defined as:
   -9999.9 Missing value

color (4-byte float, array size: npixel x nscan):
The RMSE of the chosen profiles at the surface. Values are in mm/hr. Special values are defined as:
   -9999.9 Missing value
fit (4-byte float, array size: npixel x nscan):
How well the observed Tcs match the database Tcs. It is calculated as the mean(sum(Tcobs-
Tcdtb)). Values are in K. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_2APRPSSAPHIR_H_
#define _TK_2APRPSSAPHIR_H_

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif
#endif

#ifndef _L2APRPSSAPHIR_S1_
#define _L2APRPSSAPHIR_S1_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[182];
    float Longitude[182];
    short qualityFlag[182];
    float surfacePrecipitation[182];
    float error[182];
    float fit[182];
} L2APRPSSAPHIR_S1;
#endif
#endif
```
Fortran Structure Header file:

```
STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE
```

```
STRUCTURE /L2APRPSSAPHIR_S1/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(182)
    REAL*4 Longitude(182)
    INTEGER*2 qualityFlag(182)
    REAL*4 surfacePrecipitation(182)
    REAL*4 error(182)
    REAL*4 fit(182)
END STRUCTURE
```

5.43 3GPROF - GPROF Profiling

3GPROF, "GPROF Profiling", produces global 0.25° x 0.25° gridded means using Level 2 Gprof data. Vertical hydrometeor profiles and surface rainfall means are computed. Various pixel counts are also reported. The PI is Joyce Chou. The product can be monthly or daily. The following sections describe the structure and contents of the format.

Dimension definitions:

- nlat 720 Number of 0.25° grid intervals of latitude from 90°N to 90°S.
- nlon 1440 Number of 0.25° grid intervals of longitude from 180°W to 180°E.
- nlayer 28 Number of profiling layers. The top of each layer is 0.5, 1.0, 1.5, ..., 9.5, 10.0, 11.0, ..., 18.0 km. The layer tops are heights above the earth’s surface.

Figure 470 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 470: Data Format Structure for 3GPROF, GPROF Profiling
FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

surfacePrecipitation (4-byte float, array size: nlat x nlon):
The monthly mean of the instantaneous precipitation rate at the surface for each grid. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

convectivePrecipitation (4-byte float, array size: nlat x nlon):
The monthly mean of the instantaneous convective precipitation rate at the surface for each grid. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

frozenPrecipitation (4-byte float, array size: nlat x nlon):
The monthly mean of the instantaneous frozen precipitation rate at the surface for each grid. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

rainWaterPath (4-byte float, array size: nlat x nlon):
The monthly mean of the total integrated rain water in the vertical atmospheric column. Values range from 0 to 3000 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

cloudWaterPath (4-byte float, array size: nlat x nlon):
The monthly mean of the total integrated cloud water in the vertical atmospheric column.
Values range from 0 to 3000 $kg/m^2$. Special values are defined as:

-9999.9 Missing value

**iceWaterPath** (4-byte float, array size: nlat x nlon):
The monthly mean of the total integrated ice water in the vertical atmospheric column. Values range from 0 to 3000 $kg/m^2$. Special values are defined as:

-9999.9 Missing value

**rainWater** (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the rain water content for each grid at each vertical layer. Values range from 0 to 10 $g/m^3$. Special values are defined as:

-9999.9 Missing value

**cloudWater** (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the cloud liquid water content for each grid at each vertical layer. Values range from 0 to 10 $g/m^3$. Special values are defined as:

-9999.9 Missing value

**cloudIce** (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the cloud ice liquid water content for each grid at each vertical layer. Values range from 0 to 10 $g/m^3$. Special values are defined as:

-9999.9 Missing value

**snow** (4-byte float, array size: nlat x nlon x nlayer):
The monthly mean of the snow liquid water content for each grid at each vertical layer. Values range from 0 to 10 $g/m^3$. Special values are defined as:

-9999.9 Missing value

**npixTotal** (4-byte integer, array size: nlat x nlon):
The monthly number of pixels with pixelStatus equal to zero for each grid. pixelStatus equal to zero means the pixel is valid and has a retrieval. npixTotal is used to compute the monthly means described above. Values range from 0 to 10000. Special values are defined as:

-9999 Missing value

**npixPrecipitation** (4-byte integer, array size: nlat x nlon):
The monthly number of pixels with surfacePrecipitation greater than 0 for each grid. Values range from 0 to 10000. Special values are defined as:

-9999 Missing value

**surfaceTypeIndex** (4-byte integer, array size: nlat x nlon):
Indicates the type of surface (Range 0 - 99).

Codes include:

1 : Ocean
2 : Sea-Ice
(3-12 are 'land classification')
3 : Maximum Vegetation
4 : High Vegetation
5 : Moderate Vegetation
6 : Low Vegetation
7 : Minimal Vegetation
8 : Maximum Snow
9 : Moderate Snow
10 : Low Snow
11 : Minimal Snow
12 : Standing Water and Rivers
13 : Water/Land Coast Boundary
14 : Water/Ice Boundary
15 : Land/Ice Boundary
60 : Multiple surface types
-99 : Missing value

fractionQuality0 (4-byte float, array size: nlat x nlon):
The fraction of the retrieved pixels in a given grid box identified as good retrievals. For regions where there are no retrieval issues this will be 1.0. Areas with surface screening or contamination issues with questionable retrievals during the accumulation period will have values less than one and should thus be used with caution for any quantitative analysis. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

fractionQuality1 (4-byte float, array size: nlat x nlon):
The fraction of total pixels with qualityFlag equal to 1 (use with caution) for each grid. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

fractionQuality2 (4-byte float, array size: nlat x nlon):
The fraction of total pixels with qualityFlag equal to 2 (use with extreme care over snow covered surface) for each grid. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

fractionQuality3 (4-byte float, array size: nlat x nlon):
The fraction of total pixels with qualityFlag equal to 3 (use with extreme caution) for each grid. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_3GPROF_H_
define _TK_3GPROF_H_

#ifndef _L3GPROF_GRID_
define _L3GPROF_GRID_

typedef struct {
```
float surfacePrecipitation[1440][720];
float convectivePrecipitation[1440][720];
float frozenPrecipitation[1440][720];
float rainWaterPath[1440][720];
float cloudWaterPath[1440][720];
float iceWaterPath[1440][720];
float rainWater[28][1440][720];
float cloudWater[28][1440][720];
float cloudIce[28][1440][720];
float snow[28][1440][720];
int npixTotal[1440][720];
int npixPrecipitation[1440][720];
int surfaceTypeIndex[1440][720];
float fractionQuality0[1440][720];
float fractionQuality1[1440][720];
float fractionQuality2[1440][720];
float fractionQuality3[1440][720];
} L3GPROF_GRID;

#endif
#endif

Fortran Structure Header file:

STRUCTURE /L3GPROF_GRID/
  REAL*4 surfacePrecipitation(720,1440)
  REAL*4 convectivePrecipitation(720,1440)
  REAL*4 frozenPrecipitation(720,1440)
  REAL*4 rainWaterPath(720,1440)
  REAL*4 cloudWaterPath(720,1440)
  REAL*4 iceWaterPath(720,1440)
  REAL*4 rainWater(720,1440,28)
  REAL*4 cloudWater(720,1440,28)
  REAL*4 cloudIce(720,1440,28)
  REAL*4 snow(720,1440,28)
  INTEGER*4 npixTotal(720,1440)
  INTEGER*4 npixPrecipitation(720,1440)
  INTEGER*4 surfaceTypeIndex(720,1440)
  REAL*4 fractionQuality0(720,1440)
  REAL*4 fractionQuality1(720,1440)
  REAL*4 fractionQuality2(720,1440)
  REAL*4 fractionQuality3(720,1440)
3PRPSSAPHIR, "Gridded PRPS", produces global 0.25° × 0.25° gridded means using Level 2 PRPS data. Surface precipitation means are computed. Various pixel counts are also reported. The PI is Joyce Chou. The product can be monthly or daily. The following sections describe the structure and contents of the format.

Dimension definitions:

- nlat 720 Number of 0.25° grid intervals of latitude from 90°N to 90°S.
- nlon 1440 Number of 0.25° grid intervals of longitude from 180°W to 180°E.

Figure 471 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):  
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputFileNames** (Metadata):  
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.
**InputAlgorithmVersions** (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

**InputGenerationDateTimes** (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**Grid** (Grid)

**GridHeader** (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**surfacePrecipitation** (4-byte float, array size: nlat x nlon):
The monthly mean of the instantaneous precipitation rate at the surface for each grid box. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

**error** (4-byte float, array size: nlat x nlon):
The monthly mean of the L2 error values for each grid box. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**fit** (4-byte float, array size: nlat x nlon):
The monthly mean of the L2 fit values for each grid box. Values are in K. Special values are defined as:
-9999.9 Missing value

**dataQuality** (4-byte float, array size: nlat x nlon):
The monthly percent of pixels with qualityFlag equal to 0. A value of 100 means all pixels in the grid box are good. Values range from 0 to 100 percent. Special values are defined as:
-9999.9 Missing value

**npixTotal** (4-byte integer, array size: nlat x nlon):
The monthly number of pixels in each grid box. Values range from 0 to 10000. Special values are defined as:
-9999 Missing value

**npixPrecipitation** (4-byte integer, array size: nlat x nlon):
The monthly number of pixels in each grid box with surfacePrecipitation greater than 0.
Values range from 0 to 10000. Special values are defined as:
-9999  Missing value

C Structure Header file:

```c
#ifndef _TK_3PRPSSAPHIR_H_
define _TK_3PRPSSAPHIR_H_

#ifndef _L3PRPSSAPHIR_GRID_
define _L3PRPSSAPHIR_GRID_

typedef struct {
    float surfacePrecipitation[1440][720];
    float error[1440][720];
    float fit[1440][720];
    float dataQuality[1440][720];
    int npixTotal[1440][720];
    int npixPrecipitation[1440][720];
} L3PRPSSAPHIR_GRID;

#endif
#endif
```

Fortran Structure Header file:

```fortran
STRUCTURE /L3PRPSSAPHIR_GRID/
    REAL*4 surfacePrecipitation(720,1440)
    REAL*4 error(720,1440)
    REAL*4 fit(720,1440)
    REAL*4 dataQuality(720,1440)
    INTEGER*4 npixTotal(720,1440)
    INTEGER*4 npixPrecipitation(720,1440)
END STRUCTURE
```

5.45 1BKu - Ku Power

The Ku Level-1B product, 1BKu, "Ku Power," is written as a swath structure. The swath name is "NS", for Normal scan Swath. The Ka Level-1B product, 1BKa, is closely related. The scan times in 1BKa are identical to the scan times in 1BKu. The following sections describe the structure and contents of the format.
Dimension definitions:

nscan  var  Number of scans in the granule.
nray  49  Number of angle bins in each scan.
nbin  260  Number of range bins in each ray.

Figure 472 through Figure 481 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the
Figure 473: Data Format Structure for 1BKu, ScanTime

Figure 474: Data Format Structure for 1BKu, scanStatus
5.45 1BKu - Ku Power

**Figure 475:** Data Format Structure for 1BKu, navigation

- **scPos** 4 bytes Array: XYZ x nscan
- **scVel** 4 bytes Array: XYZ x nscan
- **scLat** 4 bytes Array: nscan
- **scLon** 4 bytes Array: nscan
- **scAlt** 4 bytes Array: nscan
- **dprAlt** 4 bytes Array: nscan
- **scAttRollGeoc** 4 bytes Array: nscan
- **scAttPitchGeoc** 4 bytes Array: nscan
- **scAttYawGeoc** 4 bytes Array: nscan
- **scAttRollGeod** 4 bytes Array: nscan
- **scAttPitchGeod** 4 bytes Array: nscan
- **scAttYawGeod** 4 bytes Array: nscan
- **greenHourAng** 4 bytes Array: nscan
- **timeMidScan** 8 bytes Array: nscan
- **timeMidScanOffset** 8 bytes Array: nscan

**Figure 476:** Data Format Structure for 1BKu, rayPointing

- **rayDirectionX** 4 bytes Array: nray x nscan
- **rayDirectionY** 4 bytes Array: nray x nscan
- **instrumentYaw** 4 bytes Array: nray x nscan
- **instrumentPitch** 4 bytes Array: nray x nscan
- **instrumentRoll** 4 bytes Array: nray x nscan
- **rayTiming** 4 bytes Array: nray x nscan
- **scanAngle** 4 bytes Array: nray x nscan
Figure 477: Data Format Structure for 1BKu, HouseKeeping
Figure 478: Data Format Structure for 1BKu, VertLocate

Figure 479: Data Format Structure for 1BKu, Calibration
same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**DPRKuInfo** (Metadata):
Contains DPR information. See Metadata for GPM Products for details.

**NS** (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as: -9999.9 Missing value

**scanStatus** (Group)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \(i = 1\) and other bits = 0, the unsigned integer value is \(2^i\)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCOrientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is
good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit
in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control
System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
The operational mode of KuPR/KaPR stored in science telemetry. `operationalMode` is used in `modeStatus`. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

`limitErrorFlag` (1-byte integer, array size: nscan):

- Bit flags for every ray with information about echo power limit checks.
- `limitErrorFlag` may be used in `modeStatus`.
- Detailed information is defined in L1B Product Format edited by JAXA/EORC.

`FractionalGranuleNumber` (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, `FractionalGranuleNumber = 10.5` means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:

-9999.9   Missing value

`navigation` (Group)
**scPos** (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9  Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector (m s$^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9  Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9  Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value
scAttPitchGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time.
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time.
Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order
of the components in the file is roll, pitch, and yaw. However, the angles are computed
using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll
for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic
Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the
Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity
opposite the orbit normal direction, and the X-axis is approximately in the velocity
direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
-9999.9 Missing value

scAttPitchGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time.
Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time.
Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coor-
dinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC,6 Jan 1980.
timeMidScan is used as the reference time for the scPos and scVel values. Values range
from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0
to 100 s. Special values are defined as:
-9999.9 Missing value
**rayPointing** (Group)

**rayDirectionX** (4-byte float, array size: nray x nscan):
Unit ray direction x component in mechanical coordinates. Values range from -1.0 to 1.0. Special values are defined as:
- 9999.9 Missing value

**rayDirectionY** (4-byte float, array size: nray x nscan):
Unit ray direction y component in mechanical coordinates. Values range from -1.0 to 1.0. Special values are defined as:
- 9999.9 Missing value

**instrumentYaw** (4-byte float, array size: nray x nscan):
Yaw of mechanical coordinates w.r.t. geodetic coordinates. Values range from -135 to 225 degrees. Special values are defined as:
- 9999.9 Missing value

**instrumentPitch** (4-byte float, array size: nray x nscan):
Pitch of mechanical coordinates w.r.t. geodetic coordinates. Values range from -90 to 90 degrees. Special values are defined as:
- 9999.9 Missing value

**instrumentRoll** (4-byte float, array size: nray x nscan):
Roll of mechanical coordinates w.r.t. geodetic coordinates. Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9 Missing value

**rayTiming** (4-byte float, array size: nray x nscan):
The time delay from the secondary header packet time tag to each ray (assumed as mid-time of all radar pulses for the associated rayDirection). Values range from 0 to 1.6 s. Special values are defined as:
- 9999.9 Missing value

**scanAngle** (4-byte float, array size: nray x nscan):
Angle (degrees) of the ray from nominal nadir offset about the mechanical x-axis. The sign of the angle is consistent with the sensor y-axis, i.e., the angle is positive to the right of the direction of travel if the spacecraft is in normal mode. Values range from -18 to 18 degrees. Special values are defined as:
- 9999.9 Missing value

**HouseKeeping** (Group)
rxAtt (1-byte integer, array size: nscan):  
The scan number which is determined by the L1A product. Values range from 0 to 12 dB. Special values are defined as:  
   -99  Missing value

rxAttGainOffset (4-byte float, array size: nscan):  
The actual gain of rxAtt considering the temperature dependence. Values are in dB. Special values are defined as:  
   -9999.9  Missing value

binDiffPeakDEM (2-byte integer, array size: nray x nscan):  
The number of range bins between binEchoPeak and binDEM. It is used to ensure that the VPRF is switched in accordance with the GPM satellite altitude. Values range from -260 to 260 range bin number at NS and MS, from -130 to 130 range bin number at HS respectively. Values range from -260 to 260 range bin number. Special values are defined as:  
   -9999  Missing value

scTime (8-byte float, array size: nscan):  
Scan time expressed as TAI time with and epoch of 0000Z Jan 6, 1980. This time matches the time in ScanTime. Special values are defined as:  
   -9999.9  Missing value

vprfTableVersion (1-byte integer, array size: nscan):  
The version number of VPRF table which is used in L1B process. Values range from 1 to 127 number. Special values are defined as:  
   -99  Missing value

vprfTableSelect (1-byte integer, array size: nscan):  
The selected number of VPRF table for altitude (h, km) which is used in L1B process. The range is 1 to 25.

\[
\begin{aligned}
\text{h LT 396.5} &= 1 \\
396.5 \text{ LE h LT 397.5} &= 2 \\
397.5 \text{ LE h LT 398.5} &= 3 \\
398.5 \text{ LE h LT 399.5} &= 4 \\
399.5 \text{ LE h LT 400.5} &= 5 \\
400.5 \text{ LE h LT 401.5} &= 6 \\
401.5 \text{ LE h LT 402.5} &= 7 \\
402.5 \text{ LE h LT 403.5} &= 8 \\
403.5 \text{ LE h LT 404.5} &= 9 \\
404.5 \text{ LE h LT 405.5} &= 10 \\
405.5 \text{ LE h LT 406.5} &= 11 \\
406.5 \text{ LE h LT 407.5} &= 12 \\
407.5 \text{ LE h LT 408.5} &= 13 \\
408.5 \text{ LE h LT 409.5} &= 14 \\
409.5 \text{ LE h LT 410.5} &= 15
\end{aligned}
\]
where
LT mean less than and
LE means less than or equal to

catchingInt (1-byte integer, array size: nscan):
The timing that receive window is open for the first reflected TX pulse. If catchingInt is set to 12, then the first TX pulse is received with receive window after the twelfth TX pulse. In the case of nominal operation, catchingInt is set to 12, that is, the VPRF table is used. In other cases, including GPS-status trouble, catchingInt is set 8 and limited PRF is loaded. Values range from 8 to 12 number. Special values are defined as:
- -99  Missing value

scdpFlag (1-byte integer, array size: nscan):
The side of the SCDP system and system table used.

Bit Meaning if bit=1
0  B-side is used (if bit=0, then A-side used)
1  Priority is 1 at Basic System Table. Refer to Basic System Table.
2  Priority is 2 at Basic System Table. Refer to HK telemetry.
3  Priority is 2 at Basic System Table. Refer to Basic System Table.
4  (Spare)
5  (Spare)
6  (Spare)
7  (Spare)

fcifFlag (1-byte integer, array size: nscan):
The side of FCIF system and the system table used.

Bit Meaning if bit=1
0  B-side is used (if bit=0, then A-side used)
1  Priority is 1 at Basic System Table. Refer to Basic System Table.
2  Priority is 2 at Basic System Table. Refer to HK telemetry.
3  Priority is 2 at Basic System Table. Refer to Basic System Table.
4 (Spare)
5 (Spare)
6 (Spare)
7 (Spare)

**logAmpNoiseLevel** (2-byte integer, array size: nscan):
The Noise Level at Log Amp Termination which is stored in science telemetry. Values are in counts. Special values are defined as:
-9999 Missing value

**delay** (2-byte integer, array size: nscan):
The timing offset value from space craft time in NS. In MS and HS, it is defined as offset time value from the base delay time. They are used to adjust for beam matching of along track direction. Values range from 0 to 3360 number. Special values are defined as:
-9999 Missing value

**seqCountL1A** (2-byte integer, array size: nscan):
The scan number which is determined by the L1A product. Values range from 0 to 27000 counts. Special values are defined as:
-9999 Missing value

**fcifTemp** (2-byte integer, array size: 2 x nscan):
The temperature of FCIF component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50C to 50C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

**lnaTemp** (2-byte integer, array size: 2 x nscan):
The temperature of LNA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50C to 50C.

**rdaTemp** (2-byte integer, array size: 2 x nscan):
The temperature of RDA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50C to 50C. Attenuator setting levels of Received radar antenna. Values are 0, 3, 6, 9 and 12 in dB. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

**divcomb1Temp** (2-byte integer, array size: 2 x nscan):
The temperature of divcomb2, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50C to 50C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value
**divcomb2Temp** (2-byte integer, array size: 2 x nscan):
The temperature of divcomb2, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C. Values range from -5000 to 5000 0.01 °C. Special values are defined as:
-9999 Missing value

**sspaTemp** (2-byte integer, array size: 2 x nscan):
The temperature of RDA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C. Values range from -5000 to 5000 0.01 °C. Special values are defined as:
-9999 Missing value

**rxGain** (4-byte float, array size: nray x nscan):
The total receiver gain from FCIF input to antenna input. Values are in dB. Special values are defined as:
-9999.9 Missing value

**fcifFlagAB** (1-byte integer, array size: nscan):
FCIF A-side/B-side information. This flag does not include information on the source of the decision about the fcifFlag. Special values are defined as:
-99 Missing value

**scdpFlagAB** (1-byte integer, array size: nscan):
FCIF A-side/B-side information. This flag does not include information on the source of the decision about the scdpFlag. Special values are defined as:
-99 Missing value

**VertLocate** (Group)

**landOceanFlag** (2-byte integer, array size: nray x nscan):
Land or ocean information. The values of the flag are:

0 = Water
1 = Land
2 = Coast
3 = Inland Water

**scLocalZenith** (4-byte float, array size: nray x nscan):
The angle, in degrees, between the local zenith and the beam’s center line. The local (geodetic) zenith at the intersection of the ray and the earth ellipsoid is used. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value
**startBinRange** (4-byte float, array size: nray x nscan):
The distance from the satellite to the center of the first range bin. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**echoHighResBinNumber** (2-byte integer, array size: nray x nscan):
The number of sampling without thinning out (over sampling). Range of 1-260 for NS and MS and 1-130 at HS. EDIT Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value

Meaning in Normal Mode:
0 = Over sampling range bin OR
1 = Normal sampling range bin
2 = Interpolated range bin
-99 = Outrange bin of the observation area

Meaning in internal calibration mode:
0: In internal calibration mode, this value is stored 1- 42 range bin for each ray.
-99: missing value. In internal calibration mode, this value is stored after 43 range bin for each ray as missing.

**echoLowResBinNumber** (2-byte integer, array size: nray x nscan):
The number of sampling after thinning out the normal sample. From 1 to 260 range bin number at NS and MS while from 1 to 130 at HS. Values range from 0 to 260 range bin number. Special values are defined as:
-9999 Missing value

**binEllipsoid** (2-byte integer, array size: nray x nscan):
The range bin number of the earth ellipsoid. Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value

**scRangeEllipsoid** (4-byte float, array size: nray x nscan):
The distance from instrument to ellipsoid calculated by GeoTK. Values range from 0 to 500000 m. Special values are defined as:
-9999.9 Missing value

**binDEM** (2-byte integer, array size: nray x nscan):
Range bin number of the average DEM surface elevation in a box centered on the IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS while from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing.
scRangeDEM (4-byte float, array size: nray x nscan):
The value is calculated as scRangeEllipsoid - DEMHmean second localZenithAngle. Values range from 0 to 500000 m. Special values are defined as:
-9999.9 Missing value

DEMHmean (2-byte integer, array size: nray x nscan):
Averaged DEM height, whose SRTM-30. Values range from 0 to 9000 m. Special values are defined as:
-9999 Missing value

binDEMHtop (2-byte integer, array size: nray x nscan):
The range bin number of the maximum DEM surface elevation in a box centered on the IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS, from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11 km.

binDEMHbottom (2-byte integer, array size: nray x nscan):
The range bin number of the minimum DEM surface elevation in a box centered on the IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS, from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11 km.

binEchoPeak (2-byte integer, array size: nray x nscan):
The range bin number which has maximum echoPower in each scan and each angle bin. Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value

alongTrackBeamWidth (4-byte float, array size: nray x nscan):
Radar beamwidth (degrees) at the point transmitted power reaches one half of peak power in the along-track direction.

crossTrackBeamWidth (4-byte float, array size: nray x nscan):
Radar beamwidth (degrees) at the point transmitted power reaches one half of peak power along the cross-track direction.

mainlobeEdge (2-byte integer, array size: nray x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

sidelobeRange (2-byte integer, array size: nray x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter position from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.
ellipsoidBinOffset (4-byte float, array size: nray x nscan):
The distance between center of binEllipsoid range bin and Ellipsoid position.

rangeBinSize (4-byte float, array size: nscan):
The range bin size. With VPRF, the size for NS and MS is 250.32670 m and for HS
250.32670 m. With limited PRF, the size is 250.32670 m for all three swaths.

ratioLand (1-byte integer, array size: nray x nscan):
Ratio of land area to total area in a footprint.

ratioOcean (1-byte integer, array size: nray x nscan):
Ratio of ocean area to total area in a footprint.

ratioInLand (1-byte integer, array size: nray x nscan):
Ratio of inland water area to total area in a footprint.

ratioCoast (1-byte integer, array size: nray x nscan):
Ratio of coast area to total area in a footprint.

Calibration (Group)

fcifInPower (2-byte integer, array size: nscan):
Input power value of FCIF and is set at internal calibration mode. At another mode, the
value of fcifInPower is set as missing. Values are in 0.01 dBm. Special values are defined
as:
-30000  Missing value

intAttSelect (1-byte integer, array size: nscan):
The selected number of internal attenuation that is controlled automatically with 32 steps
and is set by internal mode. At another mode, the value of fcifInPower is set as missing.
Values range from 1 to 32 step. Special values are defined as:
-99  Missing value

sspaLnaSelect (2-byte integer, array size: nscan):
In SSPA mode, sspaLnaSelect stores the number of LNA. In LNA mode, sspaLnaSelect
stores the number of SSPA. In other modes, sspaLnaSelect is given the missing value.
Values range from 1 to 128 number. Special values are defined as:
-9999  Missing value

angleBinSelect (1-byte integer, array size: nscan):
In SSPA and LNA mode, angleBinSelect contains the selected beam number. In other
operational modes, angleBinSelect is set to missing. Values range from 1 to 49 number.
Special values are defined as:
-99  Missing value
Transmitter (Group)

**radarTransPower** (4-byte float, array size: nscan): The total (sum) power of 128 SSP A elements corrected with SSP A temperature in orbit. It is based on ground test temperature data of SSP A transmission power. Special value -9999.9 for missing scan and internal calibration mode.

**transPulseWidth** (4-byte float, array size: nscan): Transmitted pulse width corrected with FCIF temperature in orbit, based on temperature test data of FCIF. Values range from 0.0000015 to 0.0000017 s. Special value -9999.9 for missing scan and internal calibration mode.

**txAntGain** (4-byte float, array size: nray x nscan): Transmitted radar antenna effectiveness (dB). Special value -9999.9 for missing scan and internal calibration mode.

Receiver (Group)

**echoCount** (1-byte char, array size: nbin x nray x nscan): The total signal count at the antenna input that includes both echo and noise power. The signal count is stored on both observation mode and calibration mode. It is basically a copy of science telemetry raw data for sampling range bins. 0 is set to both interpolated range bin and outrange bin of the observation area.

**noiseCount** (4-byte float, array size: nray x nscan): An average of the received noise count for each angle bins during suspended 4 pulses. The value -9999.9 means missing scan and internal calibration mode.

**echoPower** (2-byte integer, array size: nbin x nray x nscan): The total signal power at the antenna input that includes both echo and noise power. The numerical value of echoPower is 100 times the power expressed in dBm when the data is valid. Values between -12000 and -2000, which correspond to the power between -120 dBm and -20 dBm, are the valid values. If the echoPower is measured outside the receiving range window that depends on the pulse repetition frequency, -29999 is stored. If the data is not valid for other reasons, -30000 is stored.

Special values:
"Count value": internal calibration mode.
-29999 : Outrange bins of the observation area.
-30000 : Missing value

**noisePower** (2-byte integer, array size: nray x nscan): An average of the received noise power for each angle bins during suspended 4 pulses. Values in dBm are multiplied by 100 and stored in the file as a 2-byte integer. The unit
in the file is thus 0.01 dBm. The range is -120 dBm to -20 dBm which corresponds to values in the file from -12000 to -2000. The value -30000 means missing scan and internal calibration mode.

**noiseSampleNumber** (2-byte integer, array size: nray x nscan):
The number of noise samplings. This value is considered with frequency agility, the number of noise sampling pulse and sampling dependency, so the value is the quadruple of the value defined by the VPRF table. Values range from 0 to 1000 number. Special value -9999 for missing and internal calibration mode.

**echoSampleNumber** (1-byte integer, array size: nray x nscan):
The number of received pulse. This value is considered with frequency agility so the value is the double of the value defined by the VHRF table. Values range from 0 to 120 number. Special values are defined as:
- 48 Internal Calibration Mode
- -99 Missing scan

**rxAntGain** (4-byte float, array size: nray x nscan):
Received radar antenna effectiveness (dB). Special value -9999.9 for missing scan and internal calibration mode.

**receivedPulseWidth** (4-byte float, array size: nscan):
Received pulse width (s) after passing through band pass filter of FCIF. Special value -9999.9 for missing scan and internal calibration mode.

C Structure Header file:

```c
#ifndef _TK_1BKu_H_
#define _TK_1BKu_H_

#ifndef _L1BKu_RECEIVER_
#define _L1BKu_RECEIVER_

typedef struct {
    unsigned char echoCount[49][260];
    float noiseCount[49];
    short echoPower[49][260];
    short noisePower[49];
    short noiseSampleNumber[49];
    signed char echoSampleNumber[49];
    float rxAntGain[49];
    float receivedPulseWidth;
} L1BKu_RECEIVER;

#endif
```
#ifndef _L1BKu_TRANSMITTER_
define _L1BKu_TRANSMITTER_

typedef struct {
    float radarTransPower;
    float transPulseWidth;
    float txAntGain[49];
} L1BKu_TRANSMITTER;
#endif

#ifndef _L1BKu_CALIBRATION_
define _L1BKu_CALIBRATION_

typedef struct {
    short fcifInPower;
    signed char intAttSelect;
    short sspaLnaSelect;
    signed char angleBinSelect;
} L1BKu_CALIBRATION;
#endif

#ifndef _L1BKu_VERTLOCATE_
define _L1BKu_VERTLOCATE_

typedef struct {
    short landOceanFlag[49];
    float scLocalZenith[49];
    float startBinRange[49];
    short echoHighResBinNumber[49];
    short echoLowResBinNumber[49];
    short binEllipsoid[49];
    float scRangeEllipsoid[49];
    short binDEM[49];
    float scRangeDEM[49];
    short DEMHmean[49];
    short binDEMHtop[49];
    short binDEMHbottom[49];
    short binEchoPeak[49];
    float alongTrackBeamWidth[49];
    float crossTrackBeamWidth[49];
    short mainlobeEdge[49];
}
typedef struct {
  signed char rxAtt;
  float rxAttGainOffset;
  short binDiffPeakDEM[49];
  double scTime;
  signed char vprfTableVersion;
  signed char vprfTableSelect;
  signed char catchingInt;
  signed char scdpFlag;
  signed char fcifFlag;
  short logAmpNoiseLevel;
  short delay;
  short seqCountL1A;
  short fcifTemp[2];
  short lnaTemp[2];
  short rdaTemp[2];
  short divcomb1Temp[2];
  short divcomb2Temp[2];
  short sspaTemp[2];
  float rxGain[49];
  signed char fcifFlagAB;
  signed char scdpFlagAB;
} L1BKu_HOUSEKEEPING;

#endif

#ifndef _L1BKu_RAYPOINTING_
#define _L1BKu_RAYPOINTING_

typedef struct {
  short sidelobeRange[49];
  float ellipsoidBinOffset[49];
  float rangeBinSize;
  signed char ratioLand[49];
  signed char ratioOcean[49];
  signed char ratioInLand[49];
  signed char ratioCoast[49];
} L1BKu_VERTLOCATE;

#endif

#ifndef _L1BKu_HOUSEKEEPING_
#define _L1BKu_HOUSEKEEPING_

typedef struct {
  signed char rxAtt;
  float rxAttGainOffset;
  short binDiffPeakDEM[49];
  double scTime;
  signed char vprfTableVersion;
  signed char vprfTableSelect;
  signed char catchingInt;
  signed char scdpFlag;
  signed char fcifFlag;
  short logAmpNoiseLevel;
  short delay;
  short seqCountL1A;
  short fcifTemp[2];
  short lnaTemp[2];
  short rdaTemp[2];
  short divcomb1Temp[2];
  short divcomb2Temp[2];
  short sspaTemp[2];
  float rxGain[49];
  signed char fcifFlagAB;
  signed char scdpFlagAB;
} L1BKu_HOUSEKEEPING;

#endif

#ifndef _L1BKu_RAYPOINTING_
#define _L1BKu_RAYPOINTING_

typedef struct {
  signed char rxAtt;
  float rxAttGainOffset;
  short binDiffPeakDEM[49];
  double scTime;
  signed char vprfTableVersion;
  signed char vprfTableSelect;
  signed char catchingInt;
  signed char scdpFlag;
  signed char fcifFlag;
  short logAmpNoiseLevel;
  short delay;
  short seqCountL1A;
  short fcifTemp[2];
  short lnaTemp[2];
  short rdaTemp[2];
  short divcomb1Temp[2];
  short divcomb2Temp[2];
  short sspaTemp[2];
  float rxGain[49];
  signed char fcifFlagAB;
  signed char scdpFlagAB;
} L1BKu_HOUSEKEEPING;

#endif
typedef struct {
    float rayDirectionX[49];
    float rayDirectionY[49];
    float instrumentYaw[49];
    float instrumentPitch[49];
    float instrumentRoll[49];
    float rayTiming[49];
    float scanAngle[49];
} L1BKu_RAYPOINTING;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;
#endif

#ifndef _L1BKu_SCANSTATUS_
#define _L1BKu_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
} L1BKu_SCANSTATUS;
#endif
typedef struct {
  short Year;
  signed char Month;
  signed char DayOfMonth;
  signed char Hour;
  signed char Minute;
  signed char Second;
  short MilliSecond;
  short DayOfYear;
  double SecondOfDay;
} SCANTIME;

typedef struct {
  SCANTIME ScanTime;
  float Latitude[49];
  float Longitude[49];
  L1BKu_SCANSTATUS scanStatus;
  NAVIGATION navigation;
  L1BKu_RAYPOINTING rayPointing;
  L1BKu_HOUSEKEEPING HouseKeeping;
  L1BKu_VERTLOCATE VertLocate;
  L1BKu_CALIBRATION Calibration;
} L1BKu_NS;
L1BKu_TRANSMITTER Transmitter;
L1BKu_RECEIVER Receiver;
} L1BKu_NS;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L1BKu_RECEIVER/
    CHARACTER echoCount(260,49)
    REAL*4 noiseCount(49)
    INTEGER*2 echoPower(260,49)
    INTEGER*2 noisePower(49)
    INTEGER*2 noiseSampleNumber(49)
    BYTE echoSampleNumber(49)
    REAL*4 rxAntGain(49)
    REAL*4 receivedPulseWidth
END STRUCTURE

STRUCTURE /L1BKu_TRANSMITTER/
    REAL*4 radarTransPower
    REAL*4 transPulseWidth
    REAL*4 txAntGain(49)
END STRUCTURE

STRUCTURE /L1BKu_CALIBRATION/
    INTEGER*2 fcifInPower
    BYTE intAttSelect
    INTEGER*2 sspaLnaSelect
    BYTE angleBinSelect
END STRUCTURE

STRUCTURE /L1BKu_VERTLOCATE/
    INTEGER*2 landOceanFlag(49)
    REAL*4 scLocalZenith(49)
    REAL*4 startBinRange(49)
    INTEGER*2 echoHighResBinNumber(49)
    INTEGER*2 echoLowResBinNumber(49)
    INTEGER*2 binEllipsoid(49)
    REAL*4 scRangeEllipsoid(49)
    INTEGER*2 binDEM(49)
REAL*4 scRangeDEM(49)
INTEGER*2 DEMHmean(49)
INTEGER*2 binDEMHtop(49)
INTEGER*2 binDEMHbottom(49)
INTEGER*2 binEchoPeak(49)
REAL*4 alongTrackBeamWidth(49)
REAL*4 crossTrackBeamWidth(49)
INTEGER*2 mainlobeEdge(49)
INTEGER*2 sidelobeRange(49)
REAL*4 ellipsoidBinOffset(49)
REAL*4 rangeBinSize
BYTE ratioLand(49)
BYTE ratioOcean(49)
BYTE ratioInLand(49)
BYTE ratioCoast(49)
END STRUCTURE

STRUCTURE /L1BKu_HOUSEKEEPING/
 BYTE rxAtt
 REAL*4 rxAttGainOffset
 INTEGER*2 binDiffPeakDEM(49)
 REAL*8 scTime
 BYTE vprfTableVersion
 BYTE vprfTableSelect
 BYTE catchingInt
 BYTE scdpFlag
 BYTE fcifFlag
 INTEGER*2 logAmpNoiseLevel
 INTEGER*2 delay
 INTEGER*2 seqCountL1A
 INTEGER*2 fcifTemp(2)
 INTEGER*2 lnaTemp(2)
 INTEGER*2 rdaTemp(2)
 INTEGER*2 divcomb1Temp(2)
 INTEGER*2 divcomb2Temp(2)
 INTEGER*2 sspaTemp(2)
 REAL*4 rxGain(49)
 BYTE fcifFlagAB
 BYTE scdpFlagAB
END STRUCTURE

STRUCTURE /L1BKu_RAYPOINTING/
 REAL*4 rayDirectionX(49)
REAL*4 rayDirectionY(49)
REAL*4 instrumentYaw(49)
REAL*4 instrumentPitch(49)
REAL*4 instrumentRoll(49)
REAL*4 rayTiming(49)
REAL*4 scanAngle(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
    REAL*4 scPos(3)
    REAL*4 scVel(3)
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 dprAlt
    REAL*4 scAttRollGeoc
    REAL*4 scAttPitchGeoc
    REAL*4 scAttYawGeoc
    REAL*4 scAttRollGeod
    REAL*4 scAttPitchGeod
    REAL*4 scAttYawGeod
    REAL*4 greenHourAng
    REAL*8 timeMidScan
    REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1BKu_SCANSTATUS/
    BYTE dataQuality
    BYTE dataWarning
    BYTE missing
    BYTE modeStatus
    INTEGER*2 geoError
    INTEGER*2 geoWarning
    INTEGER*2 SCorientation
    INTEGER*2 pointingStatus
    BYTE acsModeMidScan
    BYTE targetSelectionMidScan
    BYTE operationalMode
    BYTE limitErrorFlag
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
INTEGER*2 Year
BYTE Month
BYTE DayOfMonth
BYTE Hour
BYTE Minute
BYTE Second
INTEGER*2 MilliSecond
INTEGER*2 DayOfYear
REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BKu_NS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
  REAL*4 Longitude(49)
  RECORD /L1BKu_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L1BKu_RAYPOINTING/ rayPointing
  RECORD /L1BKu_HOUSEKEEPING/ HouseKeeping
  RECORD /L1BKu_VERTLOCATE/ VertLocate
  RECORD /L1BKu_CALIBRATION/ Calibration
  RECORD /L1BKu_TRANSMITTER/ Transmitter
  RECORD /L1BKu_RECEIVER/ Receiver
END STRUCTURE

5.46  1BKa - Ka Power

The Ka Level-1B product, 1BKa, "Ka Power," is written as a two-swath structure. The
first swath is MS for Matched beam scan Swath. MS contains rays that match the middle
25 1BKu rays in location. The second swath is HS for High sensitivity beam scan Swath.
HS contains high sensitivity rays which are close to the middle 25 1BKu rays in location.
The Ku Level-1B product, 1BKu, is closely related. The scanTime in 1BKu is identical
to the scanTime in 1BKa. The following sections describe the structure and contents of
the format.

Dimension definitions:
  nscan    var  Number of scans in the granule.
  nrayMS   25   Number of rays (angle bins) in each Matched scan.
  nbinMS   260  Number of range bins in each Matched angle bin.
  nrayHS   24   Number of rays (angle bins) in each High Sensitivity scan.
  nbinHS   130  Number of range bins in each High Sensitivity angle bin.

Figure 482 through Figure 502 show the structure of this product. The text below
describes the contents of objects in the structure, the C Structure Header File and the
Figure 482: Data Format Structure for 1BKa, Ka Power

Fortran Structure Header File.
Figure 483: Data Format Structure for 1BKa, MS

Figure 484: Data Format Structure for 1BKa, HS
5.46 1BKa - Ka Power

Figure 485: Data Format Structure for 1BKa, MS, ScanTime

Figure 486: Data Format Structure for 1BKa, MS, scanStatus
Figure 487: Data Format Structure for 1BKa, MS, navigation

Figure 488: Data Format Structure for 1BKa, MS, rayPointing
Figure 489: Data Format Structure for 1BKa, MS, HouseKeeping
Figure 490: Data Format Structure for 1BKa, MS, VertLocate

Figure 491: Data Format Structure for 1BKa, MS, Calibration
Figure 492: Data Format Structure for 1BKa, MS, Transmitter

Figure 493: Data Format Structure for 1BKa, MS, Receiver
**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**DPRKaInfo** (Metadata):
Contains DPR information. See Metadata for GPM Products for details.

**MS** (Swath)
Figure 495: Data Format Structure for 1BKa, HS, scanStatus

Figure 496: Data Format Structure for 1BKa, HS, navigation
Figure 497: Data Format Structure for 1BKa, HS, rayPointing

Figure 498: Data Format Structure for 1BKa, HS, HouseKeeping
Figure 499: Data Format Structure for 1BKa, HS, VertLocate

Figure 500: Data Format Structure for 1BKa, HS, Calibration
MS_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in MS)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: nrayMS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: nrayMS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scanStatus (Group in MS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

Bit  Meaning if bit = 1
0     missing
5     geoError is not zero
6     modeStatus is not zero
dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

Bit Meaning if bit = 1
0 Beam matching is abnormal
1 VPRF table is abnormal
2 Surface table is abnormal
3 geoWarning is not zero
4 Operational mode is not observation mode
5 GPS status is abnormal
6 Spare (always 0)
7 Check sum of L1A is abnormal

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

Bit Meaning if bit = 1
0 Scan is missing
1 Science telemetry packet missing
2 Science telemetry segment within packet missing
3 Science telemetry other missing
4 Housekeeping (HK) telemetry packet missing
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

Bit Meaning if bit = 1
0 Spare (always 0)
1 SCorientation not 0 or 180
2 pointingStatus not 0
3 Non-routine limitErrorFlag
4 Non-routine operationalMode (not 1 or 11)
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)
geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
</tbody>
</table>
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
</tbody>
</table>
5.46 1BKa - Ka Power

2  SUNPOINT
3  GSPM (Gyro-less Sun Point)
4  MSM (Mission Science Mode)
5  SLEW
6  DELTAH
7  DELTAV
-99  UNKNOWN -- ACS mode unavailable

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
</tbody>
</table>
17  Ku/Ka Independent Standby VPRF Table OUT
18  Ku/Ka Independent Standby Phase Out
19  Ku/Ka Independent Standby Dump Out
20  Ku/Ka Independent Standby (No Science Data)

**limitErrorFlag** (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. 
limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**navigation** (Group in MS)

**scPos** (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector (m/s) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9  Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
-9999.9  Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):  
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:  
-9999.9  Missing value

**scAttYawGeod** (4-byte float, array size: nscan):  
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:  
-9999.9  Missing value

**greenHourAng** (4-byte float, array size: nscan):  
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:  
-9999.9  Missing value

**timeMidScan** (8-byte float, array size: nscan):  
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. TimeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:  
-9999.9  Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):  
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:  
-9999.9  Missing value

---

**rayPointing** (Group in MS)

**rayDirectionX** (4-byte float, array size: nrayMS x nscan):  
Unit ray direction x component in mechanical coordinates. Values range from -1.0 to 1.0. Special values are defined as:  
-9999.9  Missing value

**rayDirectionY** (4-byte float, array size: nrayMS x nscan):  
Unit ray direction y component in mechanical coordinates. Values range from -1.0 to 1.0. Special values are defined as:  
-9999.9  Missing value

**instrumentYaw** (4-byte float, array size: nrayMS x nscan):  
Yaw of mechanical coordinates w.r.t. geodetic coordinates. Values range from -135 to 225 degrees. Special values are defined as:  
-9999.9  Missing value
**instrumentPitch** (4-byte float, array size: nrayMS x nscan):
Pitch of mechanical coordinates w.r.t. geodetic coordinates. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**instrumentRoll** (4-byte float, array size: nrayMS x nscan):
Roll of mechanical coordinates w.r.t. geodetic coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**rayTiming** (4-byte float, array size: nrayMS x nscan):
The time delay from the secondary header packet time tag to each ray (assumed as mid-time of all radar pulses for the associated rayDirection). Values range from 0 to 1.6 s. Special values are defined as:
-9999.9  Missing value

**scanAngle** (4-byte float, array size: nrayMS x nscan):
Angle (degrees) of the ray from nominal nadir offset about the mechanical x-axis. The sign of the angle is consistent with the sensor y-axis, i.e., the angle is positive to the right of the direction of travel if the spacecraft is in normal mode. Values range from -18 to 18 degrees. Special values are defined as:
-9999.9  Missing value

**HouseKeeping** (Group in MS)

**rxAtt** (1-byte integer, array size: nscan):
The scan number which is determined by the L1A product. Values range from 0 to 12 dB. Special values are defined as:
-99  Missing value

**rxAttGainOffset** (4-byte float, array size: nscan):
The actual gain of rxAtt considering the temperature dependence. Values are in dB. Special values are defined as:
-9999.9  Missing value

**binDiffPeakDEM** (2-byte integer, array size: nrayMS x nscan):
The number of range bins between binEchoPeak and binDEM. It is used to ensure that the VPRF is switched in accordance with the GPM satellite altitude. Values range from -260 to 260 range bin number at NS and MS, from -130 to 130 range bin number at HS respectively. Values range from -260 to 260 range bin number. Special values are defined as:
-9999  Missing value

**scTime** (8-byte float, array size: nscan):
Scan time expressed as TAI time with and epoch of 0000Z Jan 6, 1980. This time matches
the time in ScanTime. Special values are defined as:
-9999.9  Missing value

**vprfTableVersion** (1-byte integer, array size: nscan):
The version number of VPRF table which is used in L1B process. Values range from 1 to 127 number. Special values are defined as:
-99  Missing value

**vprfTableSelect** (1-byte integer, array size: nscan):
The selected number of VPRF table for altitude (h, km) which is used in L1B process. The range is 1 to 25.

\[
\begin{align*}
h & \LT 396.5 = 1 \\
396.5 \LE h & \LT 397.5 = 2 \\
397.5 \LE h & \LT 398.5 = 3 \\
398.5 \LE h & \LT 399.5 = 4 \\
399.5 \LE h & \LT 400.5 = 5 \\
400.5 \LE h & \LT 401.5 = 6 \\
401.5 \LE h & \LT 402.5 = 7 \\
402.5 \LE h & \LT 403.5 = 8 \\
403.5 \LE h & \LT 404.5 = 9 \\
404.5 \LE h & \LT 405.5 = 10 \\
405.5 \LE h & \LT 406.5 = 11 \\
406.5 \LE h & \LT 407.5 = 12 \\
407.5 \LE h & \LT 408.5 = 13 \\
408.5 \LE h & \LT 409.5 = 14 \\
409.5 \LE h & \LT 410.5 = 15 \\
410.5 \LE h & \LT 411.5 = 16 \\
411.5 \LE h & \LT 412.5 = 17 \\
412.5 \LE h & \LT 413.5 = 18 \\
413.5 \LE h & \LT 414.5 = 19 \\
414.5 \LE h & \LT 415.5 = 20 \\
415.5 \LE h & \LT 416.5 = 21 \\
416.5 \LE h & \LT 417.5 = 22 \\
417.5 \LE h & \LT 418.5 = 23 \\
418.5 \LE h & \LT 419.5 = 24 \\
419.5 \LE h & = 25
\end{align*}
\]

where
LT mean less than and
LE means less than or equal to

**catchingInt** (1-byte integer, array size: nscan):
The timing that receive window is open for the first reflected TX pulse. If catchingInt is set to 12, then the first TX pulse is received with receive window after the twelfth TX
pulse. In the case of nominal operation, catchingInt is set to 12, that is, the VPRF table is used. In other cases, including GPS-status trouble, catchingInt is set 8 and limited PRF is loaded. Values range from 8 to 12 number. Special values are defined as:
-99 Missing value

**scdpFlag** (1-byte integer, array size: nscan):
The side of the SCDF system and system table used.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit=1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>B-side is used (if bit=0, then A-side used)</td>
</tr>
<tr>
<td>1</td>
<td>Priority is 1 at Basic System Table. Refer to Basic System Table.</td>
</tr>
<tr>
<td>2</td>
<td>Priority is 2 at Basic System Table. Refer to HK telemetry.</td>
</tr>
<tr>
<td>3</td>
<td>Priority is 2 at Basic System Table. Refer to Basic System Table.</td>
</tr>
<tr>
<td>4</td>
<td>(Spare)</td>
</tr>
<tr>
<td>5</td>
<td>(Spare)</td>
</tr>
<tr>
<td>6</td>
<td>(Spare)</td>
</tr>
<tr>
<td>7</td>
<td>(Spare)</td>
</tr>
</tbody>
</table>

**fcifFlag** (1-byte integer, array size: nscan):
The side of FCIF system and the system table used.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit=1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>B-side is used (if bit=0, then A-side used)</td>
</tr>
<tr>
<td>1</td>
<td>Priority is 1 at Basic System Table. Refer to Basic System Table.</td>
</tr>
<tr>
<td>2</td>
<td>Priority is 2 at Basic System Table. Refer to HK telemetry.</td>
</tr>
<tr>
<td>3</td>
<td>Priority is 2 at Basic System Table. Refer to Basic System Table.</td>
</tr>
<tr>
<td>4</td>
<td>(Spare)</td>
</tr>
<tr>
<td>5</td>
<td>(Spare)</td>
</tr>
<tr>
<td>6</td>
<td>(Spare)</td>
</tr>
<tr>
<td>7</td>
<td>(Spare)</td>
</tr>
</tbody>
</table>

**logAmpNoiseLevel** (2-byte integer, array size: nscan):
The Noise Level at Log Amp Termination which is stored in science telemetry. Values are in counts. Special values are defined as:
-9999 Missing value

**delay** (2-byte integer, array size: nscan):
The timing offset value from spacecraft time in NS. In MS and HS, it is defined as offset time value from the base delay time. They are used to adjust for beam matching of along track direction. Values range from 0 to 3360 number. Special values are defined as:
-9999 Missing value

**seqCountL1A** (2-byte integer, array size: nscan):
The scan number which is determined by the L1A product. Values range from 0 to 27000 counts. Special values are defined as:
-9999 Missing value
fcifTemp (2-byte integer, array size: 2 x nscan):
The temperature of FCIF component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

lnaTemp (2-byte integer, array size: 2 x nscan):
The temperature of LNA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C.

divcomb2Temp (2-byte integer, array size: 2 x nscan):
The temperature of divcomb2, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C.

rdaTemp (2-byte integer, array size: 2 x nscan):
The temperature of RDA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C. Attenuator setting levels of Received radar antenna. Values are 0, 3, 6, 9 and 12 in dB. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

rxGain (4-byte float, array size: mrays x nscan):
The total receiver gain from FCIF input to antenna input. Values are in dB. Special values are defined as:
-9999.9 Missing value
fcifFlagAB (1-byte integer, array size: nscan):
FCIF A-side/B-side information. This flag does not include information on the source of
the decision about the fcifFlag. Special values are defined as:
-99 Missing value

scdpFlagAB (1-byte integer, array size: nscan):
FCIF A-side/B-side information. This flag does not include information on the source of
the decision about the scdpFlag. Special values are defined as:
-99 Missing value

VertLocate (Group in MS)

landOceanFlag (2-byte integer, array size: nrayMS x nscan):
Land or ocean information. The values of the flag are:

0 = Water
1 = Land
2 = Coast
3 = Inland Water

scLocalZenith (4-byte float, array size: nrayMS x nscan):
The angle, in degrees, between the local zenith and the beam’s center line. The local
(geodetic) zenith at the intersection of the ray and the earth ellipsoid is used. Values
range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

startBinRange (4-byte float, array size: nrayMS x nscan):
The distance from the satellite to the center of the first range bin. Values range from
350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

echoHighResBinNumber (2-byte integer, array size: nrayMS x nscan):
The number of sampling without thinning out (over sampling). Range of 1-260 for NS
and MS and 1-130 at HS. EDIT Values range from 1 to 260 range bin number. Special
values are defined as:
-9999 Missing value

Meaning in Normal Mode:
0 = Over sampling range bin OR
1 = Normal sampling range bin
2 = Interpolated range bin
-99 = Outrange bin of the observation area

Meaning in internal calibration mode:
0: In internal calibration mode, this value is stored 1- 42 range bin for each
ray.
-99999: missing value. In internal calibration mode, this value is stored after 43
range bin for each ray as missing.

echoLowResBinNumber (2-byte integer, array size: nrayMS x nscan):
The number of sampling after thinning out the normal sample. From 1 to 260 range bin
number at NS and MS while from 1 to 130 at HS. Values range from 0 to 260 range bin
number. Special values are defined as:
-999999 Missing value

binEllipsoid (2-byte integer, array size: nrayMS x nscan):
The range bin number of the earth ellipsoid. Values range from 1 to 260 range bin number.
Special values are defined as:
-999999 Missing value

scRangeEllipsoid (4-byte float, array size: nrayMS x nscan):
The distance from instrument to ellipsoid calculated by GeoTK. Values range from 0 to
500000 m. Special values are defined as:
-999999.9 Missing value

binDEM (2-byte integer, array size: nrayMS x nscan):
Range bin number of the average DEM surface elevation in a box centered on the IFOV.
Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude
is 7. On the other hand, the number of pixels in the direction of longitude reference is
changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS
while from 1 to 130 at HS. Special value is -99999 for missing scan, internal calibration
mode, or in case DEM is missing.

scRangeDEM (4-byte float, array size: nrayMS x nscan):
The value is calculated as scRangeEllipsoid - DEMHmean secand(localZenithAngle). Val-
ues range from 0 to 500000 m. Special values are defined as:
-999999.9 Missing value

DEMHmean (2-byte integer, array size: nrayMS x nscan):
Averaged DEM height, whose SRTM-30. Values range from 0 to 9000 m. Special values
are defined as:
-999999 Missing value

binDEMHtop (2-byte integer, array size: nrayMS x nscan):
The range bin number of the maximum DEM surface elevation in a box centered on the
IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction
of latitude is 7. On the other hand, the number of pixels in the direction of longitude
reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number
at NS and MS, from 1 to 130 at HS. Special value is -99999 for missing scan, internal
 calibration mode, or in case DEM is missing. The first dimension is the box size, with
sizes of 5 km x 5 km and 11 km x 11km.

binDEMHbottom (2-byte integer, array size: nrayMS x nscan):
The range bin number of the minimum DEM surface elevation in a box centered on the
IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS, from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11 km.

**binEchoPeak** (2-byte integer, array size: nrayMS x nscan):
The range bin number which has maximum echoPower in each scan and each angle bin. Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value

**alongTrackBeamWidth** (4-byte float, array size: nrayMS x nscan):
Radar beamwidth (degrees) at the point transmitted power reaches one half of peak power in the along-track direction.

**crossTrackBeamWidth** (4-byte float, array size: nrayMS x nscan):
Radar beamwidth (degrees) at the point transmitted power reaches one half of peak power along the cross-track direction.

**mainlobeEdge** (2-byte integer, array size: nrayMS x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

**sidelobeRange** (2-byte integer, array size: nrayMS x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter position from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

**ellipsoidBinOffset** (4-byte float, array size: nrayMS x nscan):
The distance between center of binEllipsoid range bin and Ellipsoid position.

**rangeBinSize** (4-byte float, array size: nscan):
The range bin size. With VPRF, the size for NS and MS is 250.32670 m and for HS 250.32670 m. With limited PRF, the size is 250.32670 m for all three swaths.

**ratioLand** (1-byte integer, array size: nrayMS x nscan):
Ratio of land area to total area in a footprint.

**ratioOcean** (1-byte integer, array size: nrayMS x nscan):
Ratio of ocean area to total area in a footprint.

**ratioInLand** (1-byte integer, array size: nrayMS x nscan):
Ratio of inland water area to total area in a footprint.

**ratioCoast** (1-byte integer, array size: nrayMS x nscan):
Ratio of coast area to total area in a footprint.

**Calibration** (Group in MS)
fcifInPower (2-byte integer, array size: nscan):
Input power value of FCIF and is set at internal calibration mode. At another mode, the value of fcifInPower is set as missing. Values are in 0.01 dBm. Special values are defined as:
-30000  Missing value

intAttSelect (1-byte integer, array size: nscan):
The selected number of internal attenuation that is controlled automatically with 32 steps and is set by internal mode. At another mode, the value of fcifInPower is set as missing. Values range from 1 to 32 step. Special values are defined as:
-99  Missing value

sspaLnaSelect (2-byte integer, array size: nscan):
In SSPA mode, sspaLnaSelect stores the number of LNA. In LNA mode, sspaLnaSelect stores the number of SSPA. In other modes, sspaLnaSelect is given the missing value. Values range from 1 to 128 number. Special values are defined as:
-9999  Missing value

angleBinSelect (1-byte integer, array size: nscan):
In SSPA and LNA mode, angleBinSelect contains the selected beam number. In other operational modes, angleBinSelect is set to missing. Values range from 1 to 49 number. Special values are defined as:
-99  Missing value

Transmitter (Group in MS)

radarTransPower (4-byte float, array size: nscan):
The total (sum) power of 128 SSPA elements corrected with SSPA temperature in orbit. It is based on ground test temperature data of SSPA transmission power. Special value -9999.9 for missing scan and internal calibration mode.

transPulseWidth (4-byte float, array size: nscan):
Transmitted pulse width corrected with FCIF temperature in orbit, based on temperature test data of FCIF. Values range from 0.0000015 to 0.0000017 s. Special value -9999.9 for missing scan and internal calibration mode.

txAntGain (4-byte float, array size: nrayMS x nscan):
Transmitted radar antenna effectiveness (dB). Special value -9999.9 for missing scan and internal calibration mode.

Receiver (Group in MS)
**echoCount** (1-byte char, array size: nbinMS x nrayMS x nscan):
The total signal count at the antenna input that includes both echo and noise power. The signal count is stored on both observation mode and calibration mode. It is basically a copy of science telemetry raw data for sampling range bins. 0 is set to both interpolated range bin and outrange bin of the observation area.

**noiseCount** (4-byte float, array size: nrayMS x nscan):
An average of the received noise count for each angle bins during suspended 4 pulses. The value -9999.9 means missing scan and internal calibration mode.

**echoPower** (2-byte integer, array size: nbinMS x nrayMS x nscan):
The total signal power at the antenna input that includes both echo and noise power. The numerical value of echoPower is 100 times the power expressed in dBm when the data is valid. Values between -12000 and -2000, which correspond to the power between -120 dBm and -20 dBm, are the valid values. If the echoPower is measured outside the receiving range window that depends on the pulse repetition frequency, -29999 is stored. If the data is not valid for other reasons, -30000 is stored.

Special values:
"Count value": internal calibration mode.
-29999 : Outrange bins of the observation area.
-30000 : Missing value

**noisePower** (2-byte integer, array size: nrayMS x nscan):
An average of the received noise power for each angle bins during suspended 4 pulses. Values in dBm are multiplied by 100 and stored in the file as a 2-byte integer. The unit in the file is thus 0.01 dBm. The range is -120 dBm to -20 dBm which corresponds to values in the file from -12000 to -2000. The value -30000 means missing scan and internal calibration mode.

**noiseSampleNumber** (2-byte integer, array size: nrayMS x nscan):
The number of noise samplings. This value is considered with frequency agility, the number of noise sampling pulse and sampling dependency, so the value is the quadruple of the value defined by the VPRF table. Values range from 0 to 1000 number. Special value -9999 for missing and internal calibration mode.

**echoSampleNumber** (1-byte integer, array size: nrayMS x nscan):
The number of received pulse. This value is considered with frequency agility so the value is the double of the value defined by the VHRF table. Values range from 0 to 120 number. Special values are defined as:
48  Internal Calibration Mode
-99  Missing scan

**rxAntGain** (4-byte float, array size: nrayMS x nscan):
Received radar antenna effectiveness (dB). Special value -9999.9 for missing scan and internal calibration mode.
**receivedPulseWidth** (4-byte float, array size: nscan):
Received pulse width (s) after passing through band pass filter of FCIF. Special value -9999.9 for missing scan and internal calibration mode.

**HS** (Swath)

**HS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in HS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value
**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: nrayHS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nrayHS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in HS)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:
Bit Meaning if bit = 1
0  Scan is missing
1  Science telemetry packet missing
2  Science telemetry segment within packet missing
3  Science telemetry other missing
4  Housekeeping (HK) telemetry packet missing
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

Bit Meaning if bit = 1
0  Spare (always 0)
1  SCorientation not 0 or 180
2  pointingStatus not 0
3  Non-routine limitErrorFlag
4  Non-routine operationalMode (not 1 or 11)
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0  Latitude limit exceeded for viewed pixel locations
1  Negative scan time, invalid input
2  Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. **geoWarning** does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^{*i}$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft
axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
</tbody>
</table>
Flight Z axis nadir, -X in flight direction
+90 yaw for DPR antenna pattern calibration
-90 yaw for DPR antenna pattern calibration
-99 Missing

**operationalMode** (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

**limitErrorFlag** (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):

The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**navigation** (Group in HS)

**scPos** (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector (m s\(^{-1}\)) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the
Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value
timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980.
timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
  -9999.9  Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
  -9999.9  Missing value

rayPointing (Group in HS)

rayDirectionX (4-byte float, array size: nrayHS x nscan):
Unit ray direction x component in mechanical coordinates. Values range from -1.0 to 1.0. Special values are defined as:
  -9999.9  Missing value

rayDirectionY (4-byte float, array size: nrayHS x nscan):
Unit ray direction y component in mechanical coordinates. Values range from -1.0 to 1.0. Special values are defined as:
  -9999.9  Missing value

instrumentYaw (4-byte float, array size: nrayHS x nscan):
Yaw of mechanical coordinates w.r.t. geodetic coordinates. Values range from -135 to 225 degrees. Special values are defined as:
  -9999.9  Missing value

instrumentPitch (4-byte float, array size: nrayHS x nscan):
Pitch of mechanical coordinates w.r.t. geodetic coordinates. Values range from -90 to 90 degrees. Special values are defined as:
  -9999.9  Missing value

instrumentRoll (4-byte float, array size: nrayHS x nscan):
Roll of mechanical coordinates w.r.t. geodetic coordinates. Values range from -180 to 180 degrees. Special values are defined as:
  -9999.9  Missing value

rayTiming (4-byte float, array size: nrayHS x nscan):
The time delay from the secondary header packet time tag to each ray (assumed as mid-time of all radar pulses for the associated rayDirection). Values range from 0 to 1.6 s. Special values are defined as:
  -9999.9  Missing value

scanAngle (4-byte float, array size: nrayHS x nscan):
Angle (degrees) of the ray from nominal nadir offset about the mechanical x_axis. The sign of the angle is consistent with the sensor y-axis, i.e., the angle is positive to the right
of the direction of travel if the spacecraft is in normal mode. Values range from -18 to 18 degrees. Special values are defined as:
-9999.9 Missing value

HouseKeeping (Group in HS)

rxAtt (1-byte integer, array size: nscan):
The scan number which is determined by the L1A product. Values range from 0 to 12 dB. Special values are defined as:
-99 Missing value

rxAttGainOffset (4-byte float, array size: nscan):
The actual gain of rxAtt considering the temperature dependence. Values are in dB. Special values are defined as:
-9999.9 Missing value

binDiffPeakDEM (2-byte integer, array size: nrayHS x nscan):
The number of range bins between binEchoPeak and binDEM. It is used to ensure that the VPRF is switched in accordance with the GPM satellite altitude. Values range from -260 to 260 range bin number at NS and MS, from -130 to 130 range bin number at HS respectively. Values range from -260 to 260 range bin number. Special values are defined as:
-9999 Missing value

scTime (8-byte float, array size: nscan):
Scan time expressed as TAI time with and epoch of 0000Z Jan 6, 1980. This time matches the time in ScanTime. Special values are defined as:
-9999.9 Missing value

vprfTableVersion (1-byte integer, array size: nscan):
The version number of VPRF table which is used in L1B process. Values range from 1 to 127 number. Special values are defined as:
-99 Missing value

vprfTableSelect (1-byte integer, array size: nscan):
The selected number of VPRF table for altitude (h, km) which is used in L1B process. The range is 1 to 25.

h LT 396.5 = 1
396.5 LE h LT 397.5 = 2
397.5 LE h LT 398.5 = 3
398.5 LE h LT 399.5 = 4
399.5 LE h LT 400.5 = 5
400.5 LE h LT 401.5 = 6
401.5 LE h LT 402.5 = 7
catchingInt (1-byte integer, array size: nscan):
The timing that receive window is open for the first reflected TX pulse. If catchingInt is set to 12, then the first TX pulse is received with receive window after the twelfth TX pulse. In the case of nominal operation, catchingInt is set to 12, that is, the VPRF table is used. In other cases, including GPS-status trouble, catchingInt is set 8 and limited PRF is loaded. Values range from 8 to 12 number. Special values are defined as:
-99 Missing value

scdpFlag (1-byte integer, array size: nscan):
The side of the SCDP system and system table used.

Bit Meaning if bit=1
0 B-side is used (if bit=0, then A-side used)
1 Priority is 1 at Basic System Table. Refer to Basic System Table.
2 Priority is 2 at Basic System Table. Refer to HK telemetry.
3 Priority is 2 at Basic System Table. Refer to Basic System Table.
4 (Spare)
5 (Spare)
6 (Spare)
7 (Spare)
fcifFlag (1-byte integer, array size: nsan):
The side of FCIF system and the system table used.

Bit Meaning if bit=1
0 B-side is used (if bit=0, then A-side used)
1 Priority is 1 at Basic System Table. Refer to Basic System Table.
2 Priority is 2 at Basic System Table. Refer to HK telemetry
3 Priority is 2 at Basic System Table. Refer to Basic System Table
4 (Spare)
5 (Spare)
6 (Spare)
7 (Spare)

logAmpNoiseLevel (2-byte integer, array size: nsan):
The Noise Level at Log Amp Termination which is stored in science telemetry. Values are in counts. Special values are defined as:
-9999 Missing value

delay (2-byte integer, array size: nsan):
The timing offset value from spacecraft time in NS. In MS and HS, it is defined as offset time value from the base delay time. They are used to adjust for beam matching of along track direction. Values range from 0 to 3360 number. Special values are defined as:
-9999 Missing value

seqCountL1A (2-byte integer, array size: nsan):
The scan number which is determined by the L1A product. Values range from 0 to 27000 counts. Special values are defined as:
-9999 Missing value

fcifTemp (2-byte integer, array size: 2 x nsan):
The temperature of FCIF component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50*C to 50*C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

lnaTemp (2-byte integer, array size: 2 x nsan):
The temperature of LNA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50*C to 50*C.

ddaTemp (2-byte integer, array size: 2 x nsan):
The temperature of RDA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50*C to 50*C. Attenuator setting levels of Received radar antenna. Values are 0, 3, 6, 9 and
12 in dB. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

**divcomb1Temp** (2-byte integer, array size: 2 x nscan):
The temperature of divcomb2, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50C to 50C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

**divcomb2Temp** (2-byte integer, array size: 2 x nscan):
The temperature of divcomb2, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50C to 50C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

**sspaTemp** (2-byte integer, array size: 2 x nscan):
The temperature of RDA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50C to 50C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

**rxGain** (4-byte float, array size: nrayHS x nscan):
The total receiver gain from FCIF input to antenna input. Values are in dB. Special values are defined as:
-9999.9 Missing value

**fcifFlagAB** (1-byte integer, array size: nscan):
FCIF A-side/B-side information. This flag does not include information on the source of the decision about the fcifFlag. Special values are defined as:
-99 Missing value

**scdpFlagAB** (1-byte integer, array size: nscan):
FCIF A-side/B-side information. This flag does not include information on the source of the decision about the scdpFlag. Special values are defined as:
-99 Missing value

**VertLocate** (Group in HS)

**landOceanFlag** (2-byte integer, array size: nrayHS x nscan):
Land or ocean information. The values of the flag are:

0 = Water
1 = Land
2 = Coast
3 = Inland Water

**scLocalZenith** (4-byte float, array size: nrayHS x nsan):
The angle, in degrees, between the local zenith and the beam’s center line. The local
(geodetic) zenith at the intersection of the ray and the earth ellipsoid is used. Values
range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**startBinRange** (4-byte float, array size: nrayHS x nsan):
The distance from the satellite to the center of the first range bin. Values range from
350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**echoHighResBinNumber** (2-byte integer, array size: nrayHS x nsan):
The number of sampling without thinning out (over sampling). Range of 1-260 for NS
and MS and 1-130 at HS. EDIT Values range from 1 to 260 range bin number. Special
values are defined as:
-9999 Missing value

**echoLowResBinNumber** (2-byte integer, array size: nrayHS x nsan):
The number of sampling after thinning out the normal sample. From 1 to 260 range bin
number at NS and MS while from 1 to 130 at HS. Values range from 0 to 260 range bin
number. Special values are defined as:
-9999 Missing value

**binEllipsoid** (2-byte integer, array size: nrayHS x nsan):
The range bin number of the earth ellipsoid. Values range from 1 to 260 range bin number.
Special values are defined as:
-9999 Missing value

**scRangeEllipsoid** (4-byte float, array size: nrayHS x nsan):
The distance from instrument to ellipsoid calculated by GeoTK. Values range from 0 to
500000 m. Special values are defined as:
-9999.9 Missing value
binDEM (2-byte integer, array size: nrayHS x nscan):
Range bin number of the average DEM surface elevation in a box centered on the IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS while from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing.

scRangeDEM (4-byte float, array size: nrayHS x nscan):
The value is calculated as scRangeEllipsoid - DEMHmean secand(localZenithAngle). Values range from 0 to 500000 m. Special values are defined as:
-9999.9 Missing value

DEMHmean (2-byte integer, array size: nrayHS x nscan):
Averaged DEM height, whose SRTM-30. Values range from 0 to 9000 m. Special values are defined as:
-9999 Missing value

binDEMHtop (2-byte integer, array size: nrayHS x nscan):
The range bin number of the maximum DEM surface elevation in a box centered on the IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS, from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11km.

binDEMHbottom (2-byte integer, array size: nrayHS x nscan):
The range bin number of the minimum DEM surface elevation in a box centered on the IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS, from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11km.

binEchoPeak (2-byte integer, array size: nrayHS x nscan):
The range bin number which has maximum echoPower in each scan and each angle bin. Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value

alongTrackBeamWidth (4-byte float, array size: nrayHS x nscan):
Radar beamwidth (degrees) at the point transmitted power reaches one half of peak power in the along-track direction.

crossTrackBeamWidth (4-byte float, array size: nrayHS x nscan):
Radar beamwidth (degrees) at the point transmitted power reaches one half of peak power along the cross-track direction.
**mainlobeEdge** (2-byte integer, array size: nrayHS x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

**sidelobeRange** (2-byte integer, array size: nrayHS x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter position from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

**ellipsoidBinOffset** (4-byte float, array size: nrayHS x nscan):
The distance between center of binEllipsoid range bin and Ellipsoid position.

**rangeBinSize** (4-byte float, array size: nscan):
The range bin size. With VPRF, the size for NS and MS is 250.32670 m and for HS 250.32670 m. With limited PRF, the size is 250.32670 m for all three swaths.

**ratioLand** (1-byte integer, array size: nrayHS x nscan):
Ratio of land area to total area in a footprint.

**ratioOcean** (1-byte integer, array size: nrayHS x nscan):
Ratio of ocean area to total area in a footprint.

**ratioInLand** (1-byte integer, array size: nrayHS x nscan):
Ratio of inland water area to total area in a footprint.

**ratioCoast** (1-byte integer, array size: nrayHS x nscan):
Ratio of coast area to total area in a footprint.

**Calibration** (Group in HS)

**fcifInPower** (2-byte integer, array size: nscan):
Input power value of FCIF and is set at internal calibration mode. At another mode, the value of fcifInPower is set as missing. Values are in 0.01 dBm. Special values are defined as:
-30000 Missing value

**intAttSelect** (1-byte integer, array size: nscan):
The selected number of internal attenuation that is controlled automatically with 32 steps and is set by internal mode. At another mode, the value of fcifInPower is set as missing. Values range from 1 to 32 step. Special values are defined as:
-99 Missing value

**sspaLnaSelect** (2-byte integer, array size: nscan):
In SSPA mode, sspaLnaSelect stores the number of LNA. In LNA mode, sspaLnaSelect stores the number of SSPA. In other modes, sspaLnaSelect is given the missing value. Values range from 1 to 128 number. Special values are defined as:
-9999 Missing value
angleBinSelect (1-byte integer, array size: nscan):
In SSPA and LNA mode, angleBinSelect contains the selected beam number. In other operational modes, angleBinSelect is set to missing. Values range from 1 to 49 number. Special values are defined as:
-99  Missing value

Transmitter (Group in HS)

radarTransPower (4-byte float, array size: nscan):
The total (sum) power of 128 SSPA elements corrected with SSPA temperature in orbit. It is based on ground test temperature data of SSPA transmission power. Special value -9999.9 for missing scan and internal calibration mode.

transPulseWidth (4-byte float, array size: nscan):
Transmitted pulse width corrected with FCIF temperature in orbit, based on temperature test data of FCIF. Values range from 0.0000015 to 0.0000017 s. Special value -9999.9 for missing scan and internal calibration mode.

Receiver (Group in HS)

echoCount (1-byte char, array size: nbinHS x nrayHS x nscan):
The total signal count at the antenna input that includes both echo and noise power. The signal count is stored on both observation mode and calibration mode. It is basically a copy of science telemetry raw data for sampling range bins. 0 is set to both interpolated range bin and outrange bin of the observation area.

noiseCount (4-byte float, array size: nrayHS x nscan):
An average of the received noise count for each angle bins during suspended 4 pulses. The value -9999.9 means missing scan and internal calibration mode.

echoPower (2-byte integer, array size: nbinHS x nrayHS x nscan):
The total signal power at the antenna input that includes both echo and noise power. The numerical value of echoPower is 100 times the power expressed in dBm when the data is valid. Values between -12000 and -2000, which correspond to the power between -120 dBm and -20 dBm, are the valid values. If the echoPower is measured outside the receiving range window that depends on the pulse repetition frequency, -29999 is stored. If the data is not valid for other reasons, -30000 is stored.
Special values:
"Count value": internal calibration mode.
-29999 : Outrange bins of the observation area.
-30000 : Missing value

**noisePower** (2-byte integer, array size: nrayHS x nscan):
An average of the received noise power for each angle bins during suspended 4 pulses. Values in dBm are multiplied by 100 and stored in the file as a 2-byte integer. The unit in the file is thus 0.01 dBm. The range is -120 dBm to -20 dBm which corresponds to values in the file from -12000 to -2000. The value -30000 means missing scan and internal calibration mode.

**noiseSampleNumber** (2-byte integer, array size: nrayHS x nscan):
The number of noise samplings. This value is considered with frequency agility, the number of noise sampling pulse and sampling dependency, so the value is the quadruple of the value defined by the VPRF table. Values range from 0 to 1000 number. Special value -9999 for missing and internal calibration mode.

**echoSampleNumber** (1-byte integer, array size: nrayHS x nscan):
The number of received pulse. This value is considered with frequency agility so the value is the double of the value defined by the VHRF table. Values range from 0 to 120 number. Special values are defined as:
48 Internal Calibration Mode
-99 Missing scan

**rxAntGain** (4-byte float, array size: nrayHS x nscan):
Received radar antenna effectiveness (dB). Special value -9999.9 for missing scan and internal calibration mode.

**receivedPulseWidth** (4-byte float, array size: nscan):
Received pulse width (s) after passing through band pass filter of FCIF. Special value -9999.9 for missing scan and internal calibration mode.

### C Structure Header file:

```c
#ifndef _TK_1BKa_H_
#define _TK_1BKa_H_

#ifndef _L1BKa_HS_RECEIVER_
#define _L1BKa_HS_RECEIVER_

typedef struct {
    unsigned char echoCount[24][130];
    float noiseCount[24];
    short echoPower[24][130];
    short noisePower[24];
} _1BKa; 
#endif
#endif
```
short noiseSampleNumber[24];
signed char echoSampleNumber[24];
float rxAntGain[24];
float receivedPulseWidth;
} L1BKa_HS_RECEIVER;
#endif

#ifndef _L1BKa_HS_TRANSMITTER_
define _L1BKa_HS_TRANSMITTER_

typedef struct {
    float radarTransPower;
    float transPulseWidth;
    float txAntGain[24];
} L1BKa_HS_TRANSMITTER;
#endif

#ifndef _L1BKa_HS_CALIBRATION_
define _L1BKa_HS_CALIBRATION_

typedef struct {
    short fcifInPower;
    signed char intAttSelect;
    short sspaLnaSelect;
    signed char angleBinSelect;
} L1BKa_HS_CALIBRATION;
#endif

#ifndef _L1BKa_HS_VERTLOCATE_
define _L1BKa_HS_VERTLOCATE_

typedef struct {
    short landOceanFlag[24];
    float scLocalZenith[24];
    float startBinRange[24];
    short echoHighResBinNumber[24];
    short echoLowResBinNumber[24];
    short binEllipsoid[24];
    float scRangeEllipsoid[24];
    short binDEM[24];
float scRangeDEM[24];
short DEMHmean[24];
short binDEMHtop[24];
short binDEMHbottom[24];
short binEchoPeak[24];
float alongTrackBeamWidth[24];
float crossTrackBeamWidth[24];
short mainlobeEdge[24];
short sidelobeRange[24];
float ellipsoidBinOffset[24];
float rangeBinSize;
signed char ratioLand[24];
signed char ratioOcean[24];
signed char ratioInLand[24];
signed char ratioCoast[24];
}

} L1BKa_HS_VERTLOCATE;

#endif

#ifndef _L1BKa_HS_HOUSEKEEPING_
#define _L1BKa_HS_HOUSEKEEPING_

typedef struct {
signed char rxAtt;
float rxAttGainOffset;
short binDiffPeakDEM[24];
double scTime;
signed char vprfTableVersion;
signed char vprfTableSelect;
signed char catchingInt;
signed char scdpFlag;
signed char fcifFlag;
short logAmpNoiseLevel;
short delay;
short seqCountL1A;
short fcifTemp[2];
short lnaTemp[2];
short rdaTemp[2];
short divcomb1Temp[2];
short divcomb2Temp[2];
short sspaTemp[2];
float rxGain[24];
signed char fcifFlagAB;
signed char scdpFlagAB;
} L1BKa_HS_HOUSEKEEPING;

#endif

#ifndef _L1BKa_HS_RAYPOINTING_
define _L1BKa_HS_RAYPOINTING_

typedef struct {
    float rayDirectionX[24];
    float rayDirectionY[24];
    float instrumentYaw[24];
    float instrumentPitch[24];
    float instrumentRoll[24];
    float rayTiming[24];
    float scanAngle[24];
} L1BKa_HS_RAYPOINTING;
#endif

#ifndef _L1BKa_HSSCANSTATUS_
define _L1BKa_HSSCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L1BKa_HS_SCANSTATUS;
#endif

#ifndef _L1BKa_HS_
define _L1BKa_HS_


typedef struct {
    SCANTIME ScanTime;
    float Latitude[24];
    float Longitude[24];
    L1BKa_HS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L1BKa_HS_RAYPOINTING rayPointing;
    L1BKa_HS_HOUSEKEEPING HouseKeeping;
    L1BKa_HS_VERTLOCATE VertLocate;
    L1BKa_HS_CALIBRATION Calibration;
    L1BKa_HS_TRANSMITTER Transmitter;
    L1BKa_HS_RECEIVER Receiver;
} L1BKa_HS;
#endif

#ifndef _L1BKa_MS_RECEIVER_
#define _L1BKa_MS_RECEIVER_

typedef struct {
    unsigned char echoCount[25][260];
    float noiseCount[25];
    short echoPower[25][260];
    short noisePower[25];
    short noiseSampleNumber[25];
    signed char echoSampleNumber[25];
    float rxAntGain[25];
    float receivedPulseWidth;
} L1BKa_MS_RECEIVER;
#endif

#ifndef _L1BKa_MS_TRANSMITTER_
#define _L1BKa_MS_TRANSMITTER_

typedef struct {
    float radarTransPower;
    float transPulseWidth;
    float txAntGain[25];
} L1BKa_MS_TRANSMITTER;
#endif
#ifndef _L1BKa_MS_CALIBRATION_
define _L1BKa_MS_CALIBRATION_

typedef struct {
    short fcifInPower;
    signed char intAttSelect;
    short sspaLnaSelect;
    signed char angleBinSelect;
} L1BKa_MS_CALIBRATION;

#endif

#define _L1BKa_MS_VERTLOCATE_

typedef struct {
    short landOceanFlag[25];
    float scLocalZenith[25];
    float startBinRange[25];
    short echoHighResBinNumber[25];
    short echoLowResBinNumber[25];
    short binEllipsoid[25];
    float scRangeEllipsoid[25];
    short binDEM[25];
    float scRangeDEM[25];
    short DEMHmean[25];
    short binDEMHtop[25];
    short binDEMHbottom[25];
    short binEchoPeak[25];
    float alongTrackBeamWidth[25];
    float crossTrackBeamWidth[25];
    short mainlobeEdge[25];
    short sidelobeRange[25];
    float ellipsoidBinOffset[25];
    float rangeBinSize;
    signed char ratioLand[25];
    signed char ratioOcean[25];
    signed char ratioInLand[25];
    signed char ratioCoast[25];
} L1BKa_MS_VERTLOCATE;

#endif
typedef struct {
    signed char rxAtt;
    float rxAttGainOffset;
    short binDiffPeakDEM[25];
    double scTime;
    signed char vprfTableVersion;
    signed char vprfTableSelect;
    signed char catchingInt;
    signed char scdpFlag;
    signed char fcifFlag;
    short logAmpNoiseLevel;
    short delay;
    short seqCountL1A;
    short fcifTemp[2];
    short lnaTemp[2];
    short rdaTemp[2];
    short divcomb1Temp[2];
    short divcomb2Temp[2];
    short sspaTemp[2];
    float rxGain[25];
    signed char fcifFlagAB;
    signed char scdpFlagAB;
} L1BKa_MS_HOUSEKEEPING;

#endif

#ifndef _L1BKa_MS_RAYPOINTING_
#define _L1BKa_MS_RAYPOINTING_

typedef struct {
    float rayDirectionX[25];
    float rayDirectionY[25];
    float instrumentYaw[25];
    float instrumentPitch[25];
    float instrumentRoll[25];
    float rayTiming[25];
    float scanAngle[25];
} L1BKa_MS_RAYPOINTING;
#endif
typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L1BKa_MS_SCANSTATUS;
typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

typedef struct {
    SCANTIME ScanTime;
    float Latitude[25];
    float Longitude[25];
    L1BKa_MS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L1BKa_MS_RAYPOINTING rayPointing;
    L1BKa_MS_HOUSEKEEPING HouseKeeping;
    L1BKa_MS_VERTLOCATE VertLocate;
    L1BKa_MS_CALIBRATION Calibration;
    L1BKa_MS_TRANSMITTER Transmitter;
    L1BKa_MS_RECEIVER Receiver;
} L1BKa_MS;

typedef struct {
    L1BKa_MS_SWATHS swath;
    L1BKa_MS_BRIGHTNESS brightness;
    L1BKa_MS_PIVOT pivot;
} L1BKa_MS_SWATHS;
L1BKa_MS MS;
L1BKa_HS HS;
} L1BKa_SWATHS;

#endif
#endif

Fortran Structure Header file:

STRUCTURE /L1BKa_HS_RECEIVER/
  CHARACTER echoCount(130,24)
  REAL*4 noiseCount(24)
  INTEGER*2 echoPower(130,24)
  INTEGER*2 noisePower(24)
  INTEGER*2 noiseSampleNumber(24)
  BYTE echoSampleNumber(24)
  REAL*4 rxAntGain(24)
  REAL*4 receivedPulseWidth
END STRUCTURE

STRUCTURE /L1BKa_HS_TRANSMITTER/
  REAL*4 radarTransPower
  REAL*4 transPulseWidth
  REAL*4 txAntGain(24)
END STRUCTURE

STRUCTURE /L1BKa_HS_CALIBRATION/
  INTEGER*2 fcifInPower
  BYTE intAttSelect
  INTEGER*2 sspaLnaSelect
  BYTE angleBinSelect
END STRUCTURE

STRUCTURE /L1BKa_HS_VERTLOCATE/
  INTEGER*2 landOceanFlag(24)
  REAL*4 scLocalZenith(24)
  REAL*4 startBinRange(24)
  INTEGER*2 echoHighResBinNumber(24)
  INTEGER*2 echoLowResBinNumber(24)
  INTEGER*2 binEllipsoid(24)
  REAL*4 scRangeEllipsoid(24)
  INTEGER*2 binDEM(24)
REAL*4 scRangeDEM(24)
INTEGER*2 DEMHmean(24)
INTEGER*2 binDEMHtop(24)
INTEGER*2 binDEMHbottom(24)
INTEGER*2 binEchoPeak(24)
REAL*4 alongTrackBeamWidth(24)
REAL*4 crossTrackBeamWidth(24)
INTEGER*2 mainlobeEdge(24)
INTEGER*2 sidelobeRange(24)
REAL*4 ellipsoidBinOffset(24)
REAL*4 rangeBinSize
BYTE ratioLand(24)
BYTE ratioOcean(24)
BYTE ratioInLand(24)
BYTE ratioCoast(24)

END STRUCTURE

STRUCTURE /L1BKa_HS_HOUSEKEEPING/
 BYTE rxAtt
 REAL*4 rxAttGainOffset
 INTEGER*2 binDiffPeakDEM(24)
 REAL*8 scTime
 BYTE vprfTableVersion
 BYTE vprfTableSelect
 BYTE catchingInt
 BYTE scdpFlag
 BYTE fcifFlag
 INTEGER*2 logAmpNoiseLevel
 INTEGER*2 delay
 INTEGER*2 seqCountL1A
 INTEGER*2 fcifTemp(2)
 INTEGER*2 lnaTemp(2)
 INTEGER*2 rdaTemp(2)
 INTEGER*2 divcomb1Temp(2)
 INTEGER*2 divcomb2Temp(2)
 INTEGER*2 sspaTemp(2)
 REAL*4 rxGain(24)
 BYTE fcifFlagAB
 BYTE scdpFlagAB

END STRUCTURE

STRUCTURE /L1BKa_HS_RAYPOINTING/
 REAL*4 rayDirectionX(24)
REAL*4 rayDirectionY(24)
REAL*4 instrumentYaw(24)
REAL*4 instrumentPitch(24)
REAL*4 instrumentRoll(24)
REAL*4 rayTiming(24)
REAL*4 scanAngle(24)

END STRUCTURE

STRUCTURE /L1BKa_HS_SCANSTATUS/

BYTE dataQuality
BYTE dataWarning
BYTE missing
BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
BYTE limitErrorFlag
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /L1BKa_HS/

RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(24)
REAL*4 Longitude(24)
RECORD /L1BKa_HS_SCANSTATUS/ scanStatus
RECORD /NAVIGATION/ navigation
RECORD /L1BKa_HS_RAYPOINTING/ rayPointing
RECORD /L1BKa_HS_HOUSEKEEPING/ HouseKeeping
RECORD /L1BKa_HS_VERTLOCATE/ VertLocate
RECORD /L1BKa_HS_CALIBRATION/ Calibration
RECORD /L1BKa_HS_TRANSMITTER/ Transmitter
RECORD /L1BKa_HS_RECEIVER/ Receiver

END STRUCTURE

STRUCTURE /L1BKa_MS_RECEIVER/

CHARACTER echoCount(260,25)
REAL*4 noiseCount(25)
INTEGER*2 echoPower(260,25)
INTEGER*2 noisePower(25)
INTEGER*2 noiseSampleNumber(25)
BYTE echoSampleNumber(25)
REAL*4 rxAntGain(25)
REAL*4 receivedPulseWidth
END STRUCTURE

STRUCTURE /L1BKa_MS_TRANSMITTER/
  REAL*4 radarTransPower
  REAL*4 transPulseWidth
  REAL*4 txAntGain(25)
END STRUCTURE

STRUCTURE /L1BKa_MS_CALIBRATION/
  INTEGER*2 fcifInPower
  BYTE intAttSelect
  INTEGER*2 sspaLnaSelect
  BYTE angleBinSelect
END STRUCTURE

STRUCTURE /L1BKa_MS_VERTLOCATE/
  INTEGER*2 landOceanFlag(25)
  REAL*4 scLocalZenith(25)
  REAL*4 startBinRange(25)
  INTEGER*2 echoHighResBinNumber(25)
  INTEGER*2 echoLowResBinNumber(25)
  INTEGER*2 binEllipsoid(25)
  REAL*4 scRangeEllipsoid(25)
  INTEGER*2 binDEM(25)
  REAL*4 scRangeDEM(25)
  INTEGER*2 DEMHmean(25)
  INTEGER*2 binDEMHtop(25)
  INTEGER*2 binDEMHbottom(25)
  INTEGER*2 binEchoPeak(25)
  REAL*4 alongTrackBeamWidth(25)
  REAL*4 crossTrackBeamWidth(25)
  INTEGER*2 mainlobeBeamWidth(25)
  INTEGER*2 sidelobeRange(25)
  REAL*4 ellipsoidBinOffset(25)
  REAL*4 rangeBinSize
  BYTE ratioLand(25)
  BYTE ratioOcean(25)
  BYTE ratioInLand(25)
  BYTE ratioCoast(25)
END STRUCTURE

STRUCTURE /L1BKa_MS_HOUSEKEEPING/
  BYTE rxAtt
  REAL*4 rxAttGainOffset
  INTEGER*2 binDiffPeakDEM(25)
  REAL*8 scTime
  BYTE vprfTableVersion
  BYTE vprfTableSelect
  BYTE catchingInt
  BYTE scdpFlag
  BYTE fcifFlag
  INTEGER*2 logAmpNoiseLevel
  INTEGER*2 delay
  INTEGER*2 seqCountL1A
  INTEGER*2 fcifTemp(2)
  INTEGER*2 lnaTemp(2)
  INTEGER*2 rdaTemp(2)
  INTEGER*2 divcomb1Temp(2)
  INTEGER*2 divcomb2Temp(2)
  INTEGER*2 sspaTemp(2)
  REAL*4 rxGain(25)
  BYTE fcifFlagAB
  BYTE scdpFlagAB
END STRUCTURE

STRUCTURE /L1BKa_MS_RAYPOINTING/
  REAL*4 rayDirectionX(25)
  REAL*4 rayDirectionY(25)
  REAL*4 instrumentYaw(25)
  REAL*4 instrumentPitch(25)
  REAL*4 instrumentRoll(25)
  REAL*4 rayTiming(25)
  REAL*4 scanAngle(25)
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
  REAL*4 scAlt
  REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1BKa_MS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BKa_MS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(25)
  REAL*4 Longitude(25)
5.47 1BPR - PR Power

The PR Level-1B product, 1BPR, “PR Power,” is written as a swath structure. The swath name is "NS", for Normal scan Swath. The following sections describe the structure and contents of the format.

Dimension definitions:
- nscan var Number of scans in the granule.
- nray 49 Number of angle bins in each scan.
- nbin 260 Number of range bins in each ray. The data is observed at 250m but interpolated to 125m so the data format aligns with Ku from GPM/DPR.

Figure 503 through Figure 512 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains metadata of general interest. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1, Level 2, and Level 3 orbital data products. See Metadata for GPM Products for details.
Figure 503: Data Format Structure for 1BPR, PR Power

Figure 504: Data Format Structure for 1BPR, ScanTime
**Figure 505: Data Format Structure for 1BPR, scanStatus**

- `dataQuality` 1 byte Array: nscan
- `dataWarning` 1 byte Array: nscan
- `missing` 1 byte Array: nscan
- `modeStatus` 1 byte Array: nscan
- `geoError` 2 bytes Array: nscan
- `geoWarning` 2 bytes Array: nscan
- `SCrientation` 2 bytes Array: nscan
- `pointingStatus` 2 bytes Array: nscan
- `acsModeMidScan` 1 byte Array: nscan
- `targetSelectionMidScan` 1 byte Array: nscan
- `operationalMode` 1 byte Array: nscan
- `limitErrorFlag` 1 byte Array: nscan
- `FractionalGranuleNumber` 8 bytes Array: nscan

**Figure 506: Data Format Structure for 1BPR, navigation**

- `scPos` 4 bytes Array: XYZ x nscan
- `scVel` 4 bytes Array: XYZ x nscan
- `scLat` 4 bytes Array: nscan
- `scLon` 4 bytes Array: nscan
- `scAlt` 4 bytes Array: nscan
- `dprAlt` 4 bytes Array: nscan
- `scAttRollGeoc` 4 bytes Array: nscan
- `scAttPitchGeoc` 4 bytes Array: nscan
- `scAttYawGeoc` 4 bytes Array: nscan
- `scAttRollGeod` 4 bytes Array: nscan
- `scAttPitchGeod` 4 bytes Array: nscan
- `scAttYawGeod` 4 bytes Array: nscan
- `greenHourAng` 4 bytes Array: nscan
- `timeMidScan` 8 bytes Array: nscan
- `timeMidScanOffset` 8 bytes Array: nscan
5.47 1BPR - PR Power

**Figure 507: Data Format Structure for 1BPR, rayPointing**

- `rayDirectionX`: 4 bytes, Array: nray x nscan
- `rayDirectionY`: 4 bytes, Array: nray x nscan
- `instrumentYaw`: 4 bytes, Array: nray x nscan
- `instrumentPitch`: 4 bytes, Array: nray x nscan
- `instrumentRoll`: 4 bytes, Array: nray x nscan
- `rayTiming`: 4 bytes, Array: nray x nscan
- `scanAngle`: 4 bytes, Array: nray x nscan

**Figure 508: Data Format Structure for 1BPR, HouseKeeping**

- `rxAtt`: 1 byte, Array: nscan
- `rxAttGainOffset`: 4 bytes, Array: nscan
- `binDiffPeakDEM`: 2 bytes, Array: nray x nscan
- `scTime`: 8 bytes, Array: nscan
- `vprfTableVersion`: 1 byte, Array: nscan
- `vprfTableSelect`: 1 byte, Array: nscan
- `catchingInt`: 1 byte, Array: nscan
- `scdpFlag`: 1 byte, Array: nscan
- `fcifFlag`: 1 byte, Array: nscan
- `logAmpNoiseLevel`: 2 bytes, Array: nscan
- `delay`: 2 bytes, Array: nscan
- `seqCountL1A`: 2 bytes, Array: nscan
- `fcifTemp`: 2 bytes, Array: 2 x nscan
- `lnaTemp`: 2 bytes, Array: 2 x nscan
- `rdatemp`: 2 bytes, Array: 2 x nscan
- `divcomb1Temp`: 2 bytes, Array: 2 x nscan
- `divcomb2Temp`: 2 bytes, Array: 2 x nscan
- `sspaTemp`: 2 bytes, Array: 2 x nscan
- `rxGain`: 4 bytes, Array: nray x nscan
- `fcifFlagAB`: 1 byte, Array: nscan
- `scdpFlagAB`: 1 byte, Array: nscan
Figure 509: Data Format Structure for 1BPR, VertLocate

Figure 510: Data Format Structure for 1BPR, Calibration
5.47 1BPR - PR Power

![Data Format Structure for 1BPR, Transmitter](image)

![Data Format Structure for 1BPR, Receiver](image)

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**DPRKuInfo** (Metadata):
Contains DPR information. See Metadata for GPM Products for details.

**NS** (Swath)

**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined
as:
  -9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
  -99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
  -99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
  -99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
  -99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
  -99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
  -9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
  -9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime.sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
  -9999.9 Missing value

**Latitude** (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
  -9999.9 Missing value

**Longitude** (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
  -9999.9 Missing value
scanStatus (Group)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^{**i} \)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^{**i} \)). The non-routine situations follow:
Bit Meaning if bit = 1
0   Spare (always 0)
1   SCorientation not 0 or 180
2   pointingStatus not 0
3   Non-routine limitErrorFlag
4   Non-routine operationalMode (not 1 or 11)
5   Spare (always 0)
6   Spare (always 0)
7   Spare (always 0)

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0   Latitude limit exceeded for viewed pixel locations
1   Negative scan time, invalid input
2   Error getting spacecraft attitude at scan mid-time
3   Error getting spacecraft ephemeris at scan mid-time
4   Invalid input non-unit ray vector for any pixel
5   Ray misses Earth for any pixel with normal pointing
6   Nadir calculation error for subsatellite position
7   Pixel count with geolocation error over threshold
8   Error in getting spacecraft attitude for any pixel
9   Error in getting spacecraft ephemeris for any pixel
10  Spare (always 0)
11  Spare (always 0)
12  Spare (always 0)
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$ the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector ($v$) from the satellite forward direction of motion, measured clockwise facing down. We define $v$ in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
</tbody>
</table>
Non-nominal mission science orientation
-9999 Missing

**acsModeMidScan** (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
</tbody>
</table>
7  Ku/Ka Standby VPRF Table OUT
8  Ku/Ka Standby Phase Out
9  Ku/Ka Standby Dump Out
10 Ku/Ka Standby (No Science Data)
11 Ku/Ka Independent Observation
12 Ku/Ka Independent External Calibration
13 Ku/Ka Independent Internal Calibration
14 Ku/Ka Independent SSPA Analysis
15 Ku/Ka Independent LNA Analysis
16 Ku/Ka Independent Health-Check
17 Ku/Ka Independent Standby VPRF Table OUT
18 Ku/Ka Independent Standby Phase Out
19 Ku/Ka Independent Standby Dump Out
20 Ku/Ka Independent Standby (No Science Data)

**limitErrorFlag** (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:

-9999.9 Missing value

**navigation** (Group)

**scPos** (4-byte float, array size: XYZ x nscan):
The position vector(m) of the spacecraft in Earth-Centered Inertial (ECI) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:

-9999.9 Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector (m/s) of the spacecraft in ECI Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:

-9999.9 Missing value
**scLat** (4-byte float, array size: nscan):
The geocentric latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

**scLon** (4-byte float, array size: nscan):
The geocentric longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value
scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

rayDirectionX (4-byte float, array size: nray x nscan):
Unit ray direction x component in mechanical coordinates. Values range from -1.0 to 1.0. Special values are defined as:
-9999.9 Missing value
**rayDirectionY** (4-byte float, array size: nray x nscan):
Unit ray direction y component in mechanical coordinates. Values range from -1.0 to 1.0. Special values are defined as:
-9999.9 Missing value

**instrumentYaw** (4-byte float, array size: nray x nscan):
Yaw of mechanical coordinates w.r.t. geodetic coordinates. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**instrumentPitch** (4-byte float, array size: nray x nscan):
Pitch of mechanical coordinates w.r.t. geodetic coordinates. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**instrumentRoll** (4-byte float, array size: nray x nscan):
Roll of mechanical coordinates w.r.t. geodetic coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**rayTiming** (4-byte float, array size: nray x nscan):
The time delay from the secondary header packet time tag to each ray (assumed as mid-time of all radar pulses for the associated rayDirection). Values range from 0 to 1.6 s. Special values are defined as:
-9999.9 Missing value

**scanAngle** (4-byte float, array size: nray x nscan):
Angle (degrees) of the ray from nominal nadir offset about the mechanical x_axis. The sign of the angle is consistent with the sensor y-axis, i.e., the angle is positive to the right of the direction of travel if the spacecraft is in normal mode. Values range from -18 to 18 degrees. Special values are defined as:
-9999.9 Missing value

**HouseKeeping** (Group)

**rxAtt** (1-byte integer, array size: nscan):
The scan number which is determined by the L1A product. Values range from 0 to 12 dB. Special values are defined as:
-99 Missing value

**rxAttGainOffset** (4-byte float, array size: nscan):
The actual gain of rxAtt considering the temperature dependence. Values are in dB. Special values are defined as:
-9999.9 Missing value

**binDiffPeakDEM** (2-byte integer, array size: nray x nscan):
The number of range bins between binEchoPeak and binDEM. It is used to ensure that
the VPRF is switched in accordance with the GPM satellite altitude. Values range from -260 to 260 range bin number at NS and MS, from -130 to 130 range bin number at HS respectively. Values range from -260 to 260 range bin number. Special values are defined as:

-9999 Missing value

**scTime** (8-byte float, array size: nscan):
Scan time expressed as TAI time with and epoch of 0000Z Jan 6, 1980. This time matches the time in ScanTime. Special values are defined as:

-9999.9 Missing value

**vprfTableVersion** (1-byte integer, array size: nscan):
The version number of VPRF table which is used in L1B process. Values range from 1 to 127 number. Special values are defined as:

-99 Missing value

**vprfTableSelect** (1-byte integer, array size: nscan):
The selected number of VPRF table for altitude (h, km) which is used in L1B process. The range is 1 to 25.

\[
\begin{align*}
    h &< 396.5 = 1 \\
    396.5 &\leq h < 397.5 = 2 \\
    397.5 &\leq h < 398.5 = 3 \\
    398.5 &\leq h < 399.5 = 4 \\
    399.5 &\leq h < 400.5 = 5 \\
    400.5 &\leq h < 401.5 = 6 \\
    401.5 &\leq h < 402.5 = 7 \\
    402.5 &\leq h < 403.5 = 8 \\
    403.5 &\leq h < 404.5 = 9 \\
    404.5 &\leq h < 405.5 = 10 \\
    405.5 &\leq h < 406.5 = 11 \\
    406.5 &\leq h < 407.5 = 12 \\
    407.5 &\leq h < 408.5 = 13 \\
    408.5 &\leq h < 409.5 = 14 \\
    409.5 &\leq h < 410.5 = 15 \\
    410.5 &\leq h < 411.5 = 16 \\
    411.5 &\leq h < 412.5 = 17 \\
    412.5 &\leq h < 413.5 = 18 \\
    413.5 &\leq h < 414.5 = 19 \\
    414.5 &\leq h < 415.5 = 20 \\
    415.5 &\leq h < 416.5 = 21 \\
    416.5 &\leq h < 417.5 = 22 \\
    417.5 &\leq h < 418.5 = 23 \\
    418.5 &\leq h < 419.5 = 24 \\
    419.5 &\leq h = 25
\end{align*}
\]
where
LT mean less than and
LE means less than or equal to

catchingInt (1-byte integer, array size: nscan):
The timing that receive window is open for the first reflected TX pulse. If catchingInt is
set to 12, then the first TX pulse is received with receive window after the twelfth TX
pulse. In the case of nominal operation, catchingInt is set to 12, that is, the VPRF table
is used. In other cases, including GPS-status trouble, catchingInt is set 8 and limited
PRF is loaded. Values range from 8 to 12 number. Special values are defined as:
-99  Missing value

scdpFlag (1-byte integer, array size: nscan):
The side of the SCDP system and system table used.

Bit Meaning if bit=1
0  B-side is used (if bit=0, then A-side used)
1  Priority is 1 at Basic System Table. Refer to Basic System Table.
2  Priority is 2 at Basic System Table. Refer to HK telemetry.
3  Priority is 2 at Basic System Table. Refer to Basic System Table.
4  (Spare)
5  (Spare)
6  (Spare)
7  (Spare)

fcifFlag (1-byte integer, array size: nscan):
The side of FCIF system and the system table used.

Bit Meaning if bit=1
0  B-side is used (if bit=0, then A-side used)
1  Priority is 1 at Basic System Table. Refer to Basic System Table.
2  Priority is 2 at Basic System Table. Refer to HK telemetry
3  Priority is 2 at Basic System Table. Refer to Basic System Table
4  (Spare)
5  (Spare)
6  (Spare)
7  (Spare)

logAmpNoiseLevel (2-byte integer, array size: nscan):
The Noise Level at Log Amp Termination which is stored in science telemetry. Values
are in counts. Special values are defined as:
-9999  Missing value

delay (2-byte integer, array size: nscan):
The timing offset value from spacecraft time in NS. In MS and HS, it is defined as offset
time value from the base delay time. They are used to adjust for beam matching of along track direction. Values range from 0 to 3360 number. Special values are defined as:
-9999 Missing value

seqCountL1A (2-byte integer, array size: nscan):
The scan number which is determined by the L1A product. Values range from 0 to 27000 counts. Special values are defined as:
-9999 Missing value

fcifTemp (2-byte integer, array size: 2 x nscan):
The temperature of FCIF component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

lnaTemp (2-byte integer, array size: 2 x nscan):
The temperature of LNA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C.

rdaTemp (2-byte integer, array size: 2 x nscan):
The temperature of RDA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C. Attenuator setting levels of Received radar antenna. Values are 0, 3, 6, 9 and 12 in dB. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

divcomb1Temp (2-byte integer, array size: 2 x nscan):
The temperature of divcomb2, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

divcomb2Temp (2-byte integer, array size: 2 x nscan):
The temperature of divcomb2, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is -50°C to 50°C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

sspaTemp (2-byte integer, array size: 2 x nscan):
The temperature of RDA component, which is averaged during about 3 minutes. The first dimension is temperature and the other is the number of referenced HK telemetry. Temperature values are multiplied by 100 and stored as a 2 byte integer. The range is
-50C to 50C. Values range from -5000 to 5000 0.01 C. Special values are defined as:
-9999 Missing value

**rxGain** (4-byte float, array size: nray x nscan):
The total receiver gain from FCIF input to antenna input. Values are in dB. Special values are defined as:
-9999.9 Missing value

**fcifFlagAB** (1-byte integer, array size: nscan):
FCIF A-side/B-side information. This flag does not include information on the source of the decision about the fcifFlag. Special values are defined as:
-99 Missing value

**scdpFlagAB** (1-byte integer, array size: nscan):
FCIF A-side/B-side information. This flag does not include information on the source of the decision about the scdpFlag. Special values are defined as:
-99 Missing value

**VertLocate** (Group)

**landOceanFlag** (2-byte integer, array size: nray x nscan):
Land or ocean information. The values of the flag are:

0 = Water  
1 = Land  
2 = Coast  
3 = Inland Water

**scLocalZenith** (4-byte float, array size: nray x nscan):
The angle, in degrees, between the local zenith and the beam’s center line. The local (geodetic) zenith at the intersection of the ray and the earth ellipsoid is used. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**startBinRange** (4-byte float, array size: nray x nscan):
The distance from the satellite to the center of the first range bin. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**echoHighResBinNumber** (2-byte integer, array size: nray x nscan):
The number of sampling without thinning out (over sampling). Range of 1-260 for NS and MS and 1-130 at HS. EDIT Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value
Meaning in Normal Mode:
0 = Over sampling range bin OR
1 = Normal sampling range bin
2 = Interpolated range bin
-99 = Outrange bin of the observation area

Meaning in internal calibration mode:
0: In internal calibration mode, this value is stored 1- 42 range bin for each ray.
-99: missing value. In internal calibration mode, this value is stored after 43 range bin for each ray as missing.

**echoLowResBinNumber** (2-byte integer, array size: nray x nscan):
The number of sampling after thinning out the normal sample. From 1 to 260 range bin number at NS and MS while from 1 to 130 at HS. Values range from 0 to 260 range bin number. Special values are defined as:
-9999 Missing value

**binEllipsoid** (2-byte integer, array size: nray x nscan):
The range bin number of the earth ellipsoid. Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value

**scRangeEllipsoid** (4-byte float, array size: nray x nscan):
The distance from instrument to ellipsoid calculated by GeoTK. Values range from 0 to 500000 m. Special values are defined as:
-9999.9 Missing value

**binDEM** (2-byte integer, array size: nray x nscan):
Range bin number of the average DEM surface elevation in a box centered on the IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS while from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing.

**scRangeDEM** (4-byte float, array size: nray x nscan):
The value is calculated as scRangeEllipsoid - DEMHmean secand(localZenithAngle). Values range from 0 to 500000 m. Special values are defined as:
-9999.9 Missing value

**DEMHmean** (2-byte integer, array size: nray x nscan):
Averaged DEM height, whose SRTM-30. Values range from 0 to 9000 m. Special values are defined as:
-9999 Missing value

**binDEMHtop** (2-byte integer, array size: nray x nscan):
The range bin number of the maximum DEM surface elevation in a box centered on the IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction
of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS, from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11 km.

**binDEM** (2-byte integer, array size: nray x nscan):
The range bin number of the minimum DEM surface elevation in a box centered on the IFOV. Reference width is 5 km x 5 km. Reference number of pixels in the direction of latitude is 7. On the other hand, the number of pixels in the direction of longitude reference is changed to 21-7 by latitude. Values range from 1 to 260 range bin number at NS and MS, from 1 to 130 at HS. Special value is -9999 for missing scan, internal calibration mode, or in case DEM is missing. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11 km.

**binEchoPeak** (2-byte integer, array size: nray x nscan):
The range bin number which has maximum echoPower in each scan and each angle bin. Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value

**alongTrackBeamWidth** (4-byte float, array size: nray x nscan):
Radar beamwidth (degrees) at the point transmitted power reaches one half of peak power in the along-track direction.

**crossTrackBeamWidth** (4-byte float, array size: nray x nscan):
Radar beamwidth (degrees) at the point transmitted power reaches one half of peak power along the cross-track direction.

**mainlobeEdge** (2-byte integer, array size: nray x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

**sidelobeRange** (2-byte integer, array size: nray x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter position from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

**ellipsoidBinOffset** (4-byte float, array size: nray x nscan):
The distance between center of binEllipsoid range bin and Ellipsoid position.

**rangeBinSize** (4-byte float, array size: nscan):
The range bin size. With VPRF, the size for NS and MS is 250.32670 m and for HS 250.32670 m. With limited PRF, the size is 250.32670 m for all three swaths.

**ratioLand** (1-byte integer, array size: nray x nscan):
Ratio of land area to total area in a footprint.

**ratioOcean** (1-byte integer, array size: nray x nscan):
Ratio of ocean area to total area in a footprint.
ratioInLand (1-byte integer, array size: nray x nscan):
Ratio of inland water area to total area in a footprint.

ratioCoast (1-byte integer, array size: nray x nscan):
Ratio of coast area to total area in a footprint.

Calibration (Group)

fcifInPower (2-byte integer, array size: nscan):
Input power value of FCIF and is set at internal calibration mode. At another mode, the
value of fcifInPower is set as missing. Values are in 0.01 dBm. Special values are defined as:
-30000 Missing value

intAttSelect (1-byte integer, array size: nscan):
The selected number of internal attenuation that is controlled automatically with 32 steps
and is set by internal mode. At another mode, the value of fcifInPower is set as missing.
Values range from 1 to 32 step. Special values are defined as:
-99 Missing value

sspaLnaSelect (2-byte integer, array size: nscan):
In SSPA mode, sspaLnaSelect stores the number of LNA. In LNA mode, sspaLnaSelect
stores the number of SSPA. In other modes, sspaLnaSelect is given the missing value.
Values range from 1 to 128 number. Special values are defined as:
-9999 Missing value

angleBinSelect (1-byte integer, array size: nscan):
In SSPA and LNA mode, angleBinSelect contains the selected beam number. In other
operational modes, angleBinSelect is set to missing. Values range from 1 to 49 number.
Special values are defined as:
-99 Missing value

Transmitter (Group)

radarTransPower (4-byte float, array size: nscan):
The total (sum) power of 128 SSPA elements corrected with SSPA temperature in orbit.
It is based on ground test temperature data of SSPA transmission power. Special value
-9999.9 for missing scan and internal calibration mode.

transPulseWidth (4-byte float, array size: nscan):
Transmitted pulse width corrected with FCIF temperature in orbit, based on temperature
test data of FCIF. Values range from 0.0000015 to 0.0000017 s. Special value -9999.9 for
missing scan and internal calibration mode.
**txAntGain** (4-byte float, array size: nray x nscan):
Transmitted radar antenna effectiveness (dB). Special value -9999.9 for missing scan and internal calibration mode.

**Receiver** (Group)

**echoCount** (1-byte char, array size: nbin x nray x nscan):
The total signal count at the antenna input that includes both echo and noise power. The signal count is stored on both observation mode and calibration mode. It is basically a copy of science telemetry raw data for sampling range bins. 0 is set to both interpolated range bin and outrange bin of the observation area.

**noiseCount** (4-byte float, array size: nray x nscan):
An average of the received noise count for each angle bins during suspended 4 pulses. The value -9999.9 means missing scan and internal calibration mode.

**echoPower** (2-byte integer, array size: nbin x nray x nscan):
The total signal power at the antenna input that includes both echo and noise power. The numerical value of echoPower is 100 times the power expressed in dBm when the data is valid. Values between -12000 and -2000, which correspond to the power between -120 dBm and -20 dBm, are the valid values. If the echoPower is measured outside the receiving range window that depends on the pulse repetition frequency, -29999 is stored. If the data is not valid for other reasons, -30000 is stored.

Special values:
"Count value": internal calibration mode.
-29999 : Outrange bins of the observation area.
-30000 : Missing value

**noisePower** (2-byte integer, array size: nray x nscan):
An average of the received noise power for each angle bins during suspended 4 pulses. Values in dBm are multiplied by 100 and stored in the file as a 2-byte integer. The unit in the file is thus 0.01 dBm. The range is -120 dBm to -20 dBm which corresponds to values in the file from -12000 to -2000. The value -30000 means missing scan and internal calibration mode.

**noiseSampleNumber** (2-byte integer, array size: nray x nscan):
The number of noise samplings. This value is considered with frequency agility, the number of noise sampling pulse and sampling dependency, so the value is the quadruple of the value defined by the VPRF table. Values range from 0 to 1000 number. Special value -9999 for missing and internal calibration mode.

**echoSampleNumber** (1-byte integer, array size: nray x nscan):
The number of received pulse. This value is considered with frequency agility so the value is the double of the value defined by the VHRF table. Values range from 0 to 120 number.
Special values are defined as:
48  Internal Calibration Mode
-99  Missing scan

**rxAntGain** (4-byte float, array size: nray x nscan):
Received radar antenna effectiveness (dB). Special value -9999.9 for missing scan and internal calibration mode.

**receivedPulseWidth** (4-byte float, array size: nscan):
Received pulse width (s) after passing through band pass filter of FCIF. Special value -9999.9 for missing scan and internal calibration mode.

### C Structure Header file:

```c
#ifndef _TK_1BPR_H_
#define _TK_1BPR_H_

#ifndef _L1BPR_RECEIVER_
#define _L1BPR_RECEIVER_

typedef struct {
    unsigned char echoCount[49][260];
    float noiseCount[49];
    short echoPower[49][260];
    short noisePower[49];
    short noiseSampleNumber[49];
    signed char echoSampleNumber[49];
    float rxAntGain[49];
    float receivedPulseWidth;
} L1BPR_RECEIVER;
#endif

#endif

#ifndef _L1BPR_TRANSMITTER_
#define _L1BPR_TRANSMITTER_

typedef struct {
    float radarTransPower;
    float transPulseWidth;
    float txAntGain[49];
} L1BPR_TRANSMITTER;
#endif

#ifndef _L1BPR_CALIBRATION_
```
#define _L1BPR_CALIBRATION_

typedef struct {
    short fcifInPower;
    signed char intAttSelect;
    short sspaLnaSelect;
    signed char angleBinSelect;
} L1BPR_CALIBRATION;

#endif

#ifndef _L1BPR_VERTLOCATE_
#define _L1BPR_VERTLOCATE_
#define _L1BPR_VERTLOCATE_

typedef struct {
    short landOceanFlag[49];
    float scLocalZenith[49];
    float startBinRange[49];
    short echoHighResBinNumber[49];
    short echoLowResBinNumber[49];
    short binEllipsoid[49];
    float scRangeEllipsoid[49];
    short binDEM[49];
    float scRangeDEM[49];
    short DEMHmean[49];
    short binDEMHtop[49];
    short binDEMHbottom[49];
    short binEchoPeak[49];
    float alongTrackBeamWidth[49];
    float crossTrackBeamWidth[49];
    short mainlobeEdge[49];
    short sidelobeRange[49];
    float ellipsoidBinOffset[49];
    float rangeBinSize;
    signed char ratioLand[49];
    signed char ratioOcean[49];
    signed char ratioInLand[49];
    signed char ratioCoast[49];
} L1BPR_VERTLOCATE;

#endif

#ifndef _L1BPR_HOUSEKEEPING_
#define _L1BPR_HOUSEKEEPING_

typedef struct {
    signed char rxAtt;
    float rxAttGainOffset;
    short binDiffPeakDEM[49];
    double scTime;
    signed char vprfTableVersion;
    signed char vprfTableSelect;
    signed char catchingInt;
    signed char scdpFlag;
    signed char fcifFlag;
    short logAmpNoiseLevel;
    short delay;
    short seqCountL1A;
    short fcifTemp[2];
    short lnaTemp[2];
    short rdaTemp[2];
    short divcomb1Temp[2];
    short divcomb2Temp[2];
    short sspaTemp[2];
    float rxGain[49];
    signed char fcifFlagAB;
    signed char scdpFlagAB;
} L1BPR_HOUSEKEEPING;

#else

#define _L1BPR_RAYPOINTING_

typedef struct {
    float rayDirectionX[49];
    float rayDirectionY[49];
    float instrumentYaw[49];
    float instrumentPitch[49];
    float instrumentRoll[49];
    float rayTiming[49];
    float scanAngle[49];
} L1BPR_RAYPOINTING;

#endif
#ifndef _NAVIGATION_
define _NAVIGATION_

typedef struct {
  float scPos[3];
  float scVel[3];
  float scLat;
  float scLon;
  float scAlt;
  float dprAlt;
  float scAttRollGeoc;
  float scAttPitchGeoc;
  float scAttYawGeoc;
  float scAttRollGeod;
  float scAttPitchGeod;
  float scAttYawGeod;
  float greenHourAng;
  double timeMidScan;
  double timeMidScanOffset;
} NAVIGATION;

#endif

#ifndef _L1BPR_SCANSTATUS_
define _L1BPR_SCANSTATUS_

typedef struct {
  signed char dataQuality;
  signed char dataWarning;
  signed char missing;
  signed char modeStatus;
  short geoError;
  short geoWarning;
  short SCorientation;
  short pointingStatus;
  signed char acsModeMidScan;
  signed char targetSelectionMidScan;
  signed char operationalMode;
  signed char limitErrorFlag;
  double FractionalGranuleNumber;
} L1BPR_SCANSTATUS;

#endif
```c
#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L1BPR_NS_
define _L1BPR_NS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L1BPR_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L1BPR_RAYPOINTING rayPointing;
    L1BPRHOUSEKEEPING HouseKeeping;
    L1BPR_VERTLOCATE VertLocate;
    L1BPR_CALIBRATION Calibration;
    L1BPR_TRANSMITTER Transmitter;
    L1BPR_RECEIVER Receiver;
} L1BPR_NS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L1BPR_RECEIVER/
    CHARACTER echoCount(260,49)
```
REAL*4 noiseCount
INTEGER*2 echoPower(260,49)
INTEGER*2 noisePower(49)
INTEGER*2 noiseSampleNumber(49)
BYTE echoSampleNumber(49)
REAL*4 rxAntGain(49)
REAL*4 receivedPulseWidth

END STRUCTURE

STRUCTURE /L1BPR_TRANSMITTER/
    REAL*4 radarTransPower
    REAL*4 transPulseWidth
    REAL*4 txAntGain(49)
END STRUCTURE

STRUCTURE /L1BPR_CALIBRATION/
    INTEGER*2 fcifInPower
    BYTE intAttSelect
    INTEGER*2 sspalnaSelect
    BYTE angleBinSelect
END STRUCTURE

STRUCTURE /L1BPR_VERTLOCATE/
    INTEGER*2 landOceanFlag(49)
    REAL*4 scLocalZenith(49)
    REAL*4 startBinRange(49)
    INTEGER*2 echoHighResBinNumber(49)
    INTEGER*2 echoLowResBinNumber(49)
    INTEGER*2 binEllipsoid(49)
    REAL*4 scRangeEllipsoid(49)
    INTEGER*2 binDEM(49)
    REAL*4 scRangeDEM(49)
    INTEGER*2 DEMHmean(49)
    INTEGER*2 binDEMHtop(49)
    INTEGER*2 binDEMHbottom(49)
    INTEGER*2 binEchoPeak(49)
    REAL*4 alongTrackBeamWidth(49)
    REAL*4 crossTrackBeamWidth(49)
    INTEGER*2 mainlobeEdge(49)
    INTEGER*2 sidelobeRange(49)
    REAL*4 ellipsoidBinOffset(49)
    REAL*4 rangeBinSize
    BYTE ratioLand(49)
BYTE ratioOcean(49)
BYTE ratioInLand(49)
BYTE ratioCoast(49)
END STRUCTURE

STRUCTURE /L1BPR_HOUSEKEEPING/
  BYTE rxAtt
  REAL*4 rxAttGainOffset
  INTEGER*2 binDiffPeakDEM(49)
  REAL*8 scTime
  BYTE vprfTableVersion
  BYTE vprfTableSelect
  BYTE catchingInt
  BYTE scdpFlag
  BYTE fcifFlag
  INTEGER*2 logAmpNoiseLevel
  INTEGER*2 delay
  INTEGER*2 seqCountL1A
  INTEGER*2 fcifTemp(2)
  INTEGER*2 lnaTemp(2)
  INTEGER*2 rdaTemp(2)
  INTEGER*2 divcombiTemp(2)
  INTEGER*2 divcomb2Temp(2)
  INTEGER*2 sspaTemp(2)
  REAL*4 rxGain(49)
  BYTE fcifFlagAB
  BYTE scdpFlagAB
END STRUCTURE

STRUCTURE /L1BPR_RAYPOINTING/
  REAL*4 rayDirectionX(49)
  REAL*4 rayDirectionY(49)
  REAL*4 instrumentYaw(49)
  REAL*4 instrumentPitch(49)
  REAL*4 instrumentRoll(49)
  REAL*4 rayTiming(49)
  REAL*4 scanAngle(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
REAL*4 scLon
REAL*4 scAlt
REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L1BPR_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L1BPR_NS/
5.48 2AKu - Ku precipitation

The Ku Level-2A product, 2AKu, "Ku precipitation," is written as a 1 swath structure. The swath is NS, normal scans. The following sections describe the structure and contents of the format.

Dimension definitions:

- **nscan var** Number of scans in the granule.
- **nray** 49 Number of angle bins in each NS scan.
- **nrayMS** 25 Number of angle bins in each MS scan.
- **nrayHS** 24 Number of angle bins in each HS scan.
- **nbin** 176 Number of range bins in each NS and MS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
- **nbinSZP** 7 Number of range bins for sigmaZeroProfile.
- **nNP** 4 Number of NP kinds.
- **nearFar** 2 Near reference, Far reference.
- **foreBack** 2 Forward, Backward.
- **method** 6 Number of SRT methods.
- **nsdew** 3 Number of standard deviation effective ways.
- **nNode** 5 Number of binNode.
- **nDSD** 2 Number of DSD parameters. Parameters are dBNw and Dm (mm).
- **LS** 2 Liquid, solid.
- **nNUBF** 3 Number of NUBF parameters.

Figure 513 through Figure 524 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 513: Data Format Structure for 2AKu, Ku precipitation
### Figure 514: Data Format Structure for 2AKu, ScanTime

- **Year**: 2 bytes, Array: nscan
- **Month**: 1 byte, Array: nscan
- **DayOfMonth**: 1 byte, Array: nscan
- **Hour**: 1 byte, Array: nscan
- **Minute**: 1 byte, Array: nscan
- **Second**: 1 byte, Array: nscan
- **MilliSecond**: 2 bytes, Array: nscan
- **DayOfYear**: 2 bytes, Array: nscan
- **SecondOfDay**: 8 bytes, Array: nscan

### Figure 515: Data Format Structure for 2AKu, scanStatus

- **dataQuality**: 1 byte, Array: nscan
- **dataWarning**: 1 byte, Array: nscan
- **missing**: 1 byte, Array: nscan
- **modeStatus**: 1 byte, Array: nscan
- **geoError**: 2 bytes, Array: nscan
- **geoWarning**: 2 bytes, Array: nscan
- **SCorientation**: 2 bytes, Array: nscan
- **pointingStatus**: 2 bytes, Array: nscan
- **acsModeMidScan**: 1 byte, Array: nscan
- **targetSelectionMidScan**: 1 byte, Array: nscan
- **operationalMode**: 1 byte, Array: nscan
- **limitErrorFlag**: 1 byte, Array: nscan
- **FractionalGranuleNumber**: 8 bytes, Array: nscan
Figure 516: Data Format Structure for 2AKu, navigation
5.48 2AKu - Ku precipitation

- elevation: 4 bytes, Array: nray x nscan
- landSurfaceType: 4 bytes, Array: nray x nscan
- localZenithAngle: 4 bytes, Array: nray x nscan
- flagPrecip: 4 bytes, Array: nray x nscan
- flagSigmaZeroSaturation: 1 byte, Array: nray x nscan
- binRealSurface: 2 bytes, Array: nray x nscan
- binStormTop: 2 bytes, Array: nray x nscan
- heightStormTop: 4 bytes, Array: nray x nscan
- binClutterFreeBottom: 2 bytes, Array: nray x nscan
- sigmaZeroMeasured: 4 bytes, Array: nray x nscan
- zFactorMeasured: 4 bytes, Array: nbin x nray x nscan
- ellipsoidBinOffset: 4 bytes, Array: nray x nscan
- snRatioAtRealSurface: 4 bytes, Array: nray x nscan
- adjustFactor: 4 bytes, Array: nray x nscan
- snowIceCover: 1 byte, Array: nray x nscan

Figure 517: Data Format Structure for 2AKu, PRE

- binZeroDeg: 2 bytes, Array: nray x nscan
- attenuationNP: 4 bytes, Array: nbin x nray x nscan
- piaNP: 4 bytes, Array: nNP x nray x nscan
- sigmaZeroNPCorrected: 4 bytes, Array: nray x nscan
- heightZeroDeg: 4 bytes, Array: nray x nscan

Figure 518: Data Format Structure for 2AKu, VER
Figure 519: Data Format Structure for 2AKu, CSF

Figure 520: Data Format Structure for 2AKu, SRT
5.48 2AKu - Ku precipitation

Figure 521: Data Format Structure for 2AKu, DSD

DSD

- phase: 1 byte Array: nbin x nray x nscan
- binNode: 2 bytes Array: nNode x nray x nscan

Figure 522: Data Format Structure for 2AKu, Experimental

Experimental

- precipRateESurface2: 4 bytes Array: nray x nscan
- precipRateESurface2Status: 1 byte Array: nray x nscan
- sigmaZeroProfile: 4 bytes Array: nbinSZP x nray x nscan
- binDEML2: 2 bytes Array: nray x nscan
- seaIceConcentration: 4 bytes Array: nray x nscan
Figure 523: Data Format Structure for 2AKu, SLV

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
5.48 2AKu - Ku precipitation

FlagEcho 1 byte  Array: nbin x nray x nscan

qualityData 4 bytes  Array: nray x nscan

qualityFlag 1 byte  Array: nray x nscan

flagSensor 1 byte  Array: nscan

flagScanPattern 2 bytes  Array: nscan

Figure 524: Data Format Structure for 2AKu, FLG

FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

NS (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MillisSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

Bit Meaning if bit = 1
0 missing
5 geoError is not zero
6 modeStatus is not zero
**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
</tbody>
</table>
4  Anomalous Time Step
5  GHA not calculated due to error
6  SunData (Group) not calculated due to error
7  Failure to calculate Sun in inertial coordinates
8  Fallback to GES ephemeris
9  Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

Value Meaning
0  +X forward (yaw 0)
180 -X forward (yaw 180)
-8000 Non-nominal pointing
-9999 Missing

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

Value Meaning
0  Nominal pointing in Mission Science Mode
1  GPS point solution stale and PVT ephemeris used
2  GEONS solution stale and GEONS ephemeris used
-8000 Non-nominal mission science orientation
-9999 Missing

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

Value Meaning
0  LAUNCH
1  RATENULL
2       SUNPOINT
3       GSPM (Gyro-less Sun Point)
4       MSM (Mission Science Mode)
5       SLEW
6       DELTAH
7       DELTAV
-99     UNKNOWN -- ACS mode unavailable

**targetSelectionMidScan** (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
</tbody>
</table>
limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks.
limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPos (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector ($m s^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan):
The geocentric latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geocentric longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**PRE** (Group)

**elevation** (4-byte float, array size: nray x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:
-9999.9 Missing value

**landSurfaceType** (4-byte integer, array size: nray x nscan):
Land surface type.

0 - 99 Ocean
100 - 199 Land
200 - 299 Coast
300 - 399 Inland water
-99999 Missing value
localZenithAngle (4-byte float, array size: nray x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values
are in degree. Special values are defined as:
-9999.9 Missing value

flagPrecip (4-byte integer, array size: nray x nscan):
Precipitation or no precipitation.
For L2 Ku and L2 Ka
0 No precipitation
1 Precipitation
-9999 Missing value

For L2 DPR
0 No precipitation by both Ku and Ka
1 Precipitation by Ka, no rain by Ku
10 Precipitation by Ku, no rain by Ka
11 Precipitation by both Ku and Ka
-9999 Missing value

flagSigmaZeroSaturation (1-byte char, array size: nray x nscan):
A flag to show whether echoPower is under a saturated
level or not at a range bin with a calculation of
sigmaZeroMeasured. Values are:
0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nray x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based
ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths,
bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the
Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nray x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based

ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:

-9999  Missing value

**heightStormTop** (4-byte float, array size: nray x nscan):
Height of storm top. Values are in m. Special values are defined as:

-9999.9  Missing value

**binClutterFreeBottom** (2-byte integer, array size: nray x nscan):
Range bin number for clutter free bottom. Special values are defined as:

-9999  Missing value

**sigmaZeroMeasured** (4-byte float, array size: nray x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:

-9999.9  Missing value

**zFactorMeasured** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:

-9999.9  Missing value

**ellipsoidBinOffset** (4-byte float, array size: nray x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

\[
\text{ellipsoidBinOffset} = \text{scRangeEllipsoid} - \{ \text{startBinRange} + (\text{binEllipsoid}-1) \times \text{rangeBinSize} \}
\]

scRangeEllipsoid : Distance between a sensor and the ellipsoid [m]
startBinRange : Distance between a sensor and a center of the highest observed range bin [m]
binEllipsoid : Range bin number of the Ellipsoid (1 - 260)
rangeBinSize : Range bin size [m]

-9999  Missing value

**snRatioAtRealSurface** (4-byte float, array size: nray x nscan):
Signal/Noise ratio at real surface range bin.

\[
\text{snRatioAtRealSurface} = 10. \times \log10(\text{echoPowertrueV [mW]} / \text{noisePowertrueV [mW]})
\]

-9999  Missing value
adjustFactor (4-byte float, array size: nray x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm’) and sigmaZeroMeasured (dBs0m’).

\[ dBZm' = dBZm - \text{adjustFactor} \]
\[ dBs0m' = dBs0m - \text{adjustFactor} \]
The adjustment factor is the sum of 3 components:
- base adjustment for instrument dependency,
- angle-bin adjustment for angle-bin dependency, and
- temporal adjustment for orbit number dependency.

snowIceCover (1-byte integer, array size: nray x nscan):
TBD. Special values are defined as:
-99  Missing value

VER (Group)

binZeroDeg (2-byte integer, array size: nray x nscan):
Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

attenuationNP (4-byte float, array size: nbin x nray x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are defined as:
-9999.9  Missing value

piaNP (4-byte float, array size: nNP x nray x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are
defined as:
-9999.9  Missing value

**sigmaZeroNPCorrected** (4-byte float, array size: nray x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9  Missing value

**heightZeroDeg** (4-byte float, array size: nray x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9  Missing value

**CSF** (Group)

**flagBB** (4-byte integer, array size: nray x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.

**L2 DPR:**
0  no Bright Band
1  Bright Band detected by Ku and DFRm
2  Bright Band detected by Ku only
3  Bright Band detected by DFRm only
-1111  No rain value
-9999  Missing value

**L2 Ku and L2 Ka:**
0  BB not detected
1  BB detected
-1111  No rain value
-9999  Missing value

**binBBPeak** (2-byte integer, array size: nray x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999  Missing value

**binBBTop** (2-byte integer, array size: nray x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88
at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999  Missing value

**binBBBottom** (2-byte integer, array size: nray x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999  Missing value

**heightBB** (4-byte float, array size: nray x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9  Missing value

**widthBB** (4-byte float, array size: nray x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9  Missing value

**qualityBB** (4-byte integer, array size: nray x nscan):
Quality of the bright band. When the bright band is detected, a larger positive number indicates lower confidence in the detection.

The Ku detection is clear, but the Ka and DPR detection is somewhat doubtful.

The meaning of qualityBB has not been finalized.

3  Smeared bright band
2  Not so clear bright band
1  Clear bright band
0  BB not detected in the case of rain
-1111  No rain value
-9999  Missing value

**typePrecip** (4-byte integer, array size: nray x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, onvective, and other, can be obtained as follows:
When typePrecip is greater than zero, 
Major rain type = typePrecip/10000000
  = 1  stratiform
  = 2  convective
  = 3  other

-1111  No rain value
-9999  Missing value

Let abcdefgh be the 8 digit number,

abcdefgh
then
a: Main rain type. (a=1,2,3),
b: 0,
c: 0,
d: V rain type,
e: H rain type,
f: BB,
g: Shallow rain,
h: Small size cell.

----------------------------------------------------
The following numbers appear as Ku and Ka (MS/HS) rain types:
    ---- stratiform
    1001H100
    10031000
    ---- convective
    2001H1xy (x>0 or y>0)
    2002Hbxy
    200310xy (x>0 or y>0)
    200320xy
    ---- other
    300330xy
where H is the rain type by H-method, and b depends on BB, 
x on shallow rain and y on small size cell:
H = 1: stratiform by H-method,
    2: convective by H-method,
    3: other by H-method.

b = 0: BB not detected,
    1: BB detected.
x = 0: No shallow rain,
   1: Shallow isolated,
   3: Shallow non-isolated.

y = 0: No small size cell,
   1: Single cell,
   2: Small size cell consisting of two adjacent pixels.

In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:

DFRm rain type = (typePrecip\%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000 in FORTRAN

DFRm rain type
   = 1  stratiform
   = 2  convective
   = 4  transition
   = 8  DFRm method cannot be applicable at Part B (in this case the conventional method determines the major rain type)
   = 9  DFRm method cannot be applicable at Part A (in this case the conventional method determines the major rain type)

-1111 No rain value
-9999 Missing value

If dual frequency data is not available but Ku-only or Ka-only is available, rain type is expressed by the following 8 digit number:
   10xxxxxx --- stratiform,
   20xxxxxx --- convective,
   30xxxxxx --- other,
which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by
   1qxxxxxx --- stratiform,
   2qxxxxxx --- convective,
   3qxxxxxx --- other,
where q>0.
Thus, by examining q, users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.
For MS and HS, DFRm method is used.

DFRm decision classifies rain type into
stratiform,
convective,
and
transition.

The DPR numbering rule can be summarized as follows:
Let opqrstuv be the 8 digit number, then
- o: Main rain type. (o=1,2,3),
- p: DFRm rain type. (p=0,1,2,4,8,9, with p=0 for single frequency data only),
- q: DFRm BB. (q=0,1),
- r: V rain type (by conventional V-method).
  Basically r=0 for inner swath and r>0 for outer swath.
  However, r>0 when only single frequency data is available,
- s: H rain type,
- t: = 0 for inner swath,
  1 when BB is detected in the outer swath.
- u: Shallow rain,
- v: Small size cell.

DFRm type can be obtained by examining p

The meaning of p is as follows:
- p = 0: single frequency data only (dual frequency data not available),
  1: stratiform by DFRm method,
  2: convective by DFRm method,
  4: transition by DFRm method,
  8: DFRm decision not available,
  9: DFRm decision not available.

Note that p>0 always in DPR processing, which is different from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, p=0 always.

The following numbers appear as DPR rain types:
--- stratiform
1901H100
19031000
--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy
--- other
390330xy

***********************
* For NS inner swath and MS *
***********************
--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type (x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type (x>0 or y>0)
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type

**************
* For HS *
**************
--- stratiform
 11B0H000
 14B01000
 19001000 --- H decision only
--- convective
 21B0H0x0 (x>0)
 22B0H0x0
 240010x0 (x>0, 24B010x0 with B=0)
 240020x0
 241010x0 (x>0, 24B010x0 with B=1)
 290010x0 (x>0) --- H decision only
 290020x0 --- H decision only
--- other
 340030x0
 390030x0 --- H decision only

where w depends on BB by conventional V-method, B on BB by DFRm method, H on H-method, x on shallow rain and y on small size cell:
  w = 0: BB not detected by conventional V-method,
       1: BB detected by conventional V-method.
  B = 0: BB not detected by DFRm method,
       1: BB detected by DFRm method.
  H = 1: stratiform by H-method,
       2: convective by H-method,
       3: other by H-method.
  x = 0: No shallow rain,
       1: Shallow isolated,
       3: Shallow non-isolated.
  y = 0: No small size cell,
       1: Single cell,
2: Small size cell consisting of two adjacent pixels. In the above, x>0 and y>0 are taken care of in the function \texttt{R\_type\_classification\_dpr2()}.

\begin{verbatim}
qualityTypePrecip (4-byte integer, array size: nray x nscan):

Quality of the precipitation type.

1 Good
-1111 No rain value
-9999 Missing value

flagShallowRain (4-byte integer, array size: nray x nscan):

Type of shallow rain
0 No shallow rain
10 Shallow isolated (maybe)
11 Shallow isolated (certain)
20 Shallow non-isolated (maybe)
21 Shallow non-isolated (certain)
-1111 No rain value
-9999 Missing value

flagHeavyIcePrecip (1-byte integer, array size: nray x nscan):

This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:

-99 Missing value

flagAnvil (1-byte integer, array size: nray x nscan):

flagAnvil is 1 when anvil is detected by the Ku-band radar,
0 when anvil is not detected, and
-99 when the data is missing.

Note that Ka-band decision is not made because of a lower sensitivity of Ka-band radar (therefore, there does not exist any Ka-band flagAnvil; only Ku-band flagAnvil is available in Ku-only and DPR NS).
SRT (Group)

**PIAalt** (4-byte float, array size: method x nray x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where

- \( PIA_{alt}(j=1) = PIA_{Ku} \) from forward along-track spatial at kth angle bin
- \( PIA_{alt}(j=2) = PIA_{Ku} \) from backward along-track spatial at kth angle bin
- \( PIA_{alt}(j=3) = PIA_{Ku} \) from forward hybrid at kth angle bin
- \( PIA_{alt}(j=4) = PIA_{Ku} \) from backward hybrid at kth angle bin
- \( PIA_{alt}(j=5) = PIA_{Ku} \) from temporal reference at kth angle bin
- \( PIA_{alt}(j=6) = PIA_{Ku} \) from light-rain temporal reference at kth angle bin

Values are in dB. Special values are defined as:
- 9999.9 Missing value

**RFactorAlt** (4-byte float, array size: method x nray x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAalt. Special values are defined as:
- 9999.9 Missing value

**PIAweight** (4-byte float, array size: method x nray x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and \( \sigma_j \) is the standard deviation of reference data for method j.

\[ PIA_{weight,j} = \frac{1}{\sigma_j^2} \times \left( \frac{1}{\text{Sum}_j(1/\sigma_j^2)} \right) \]

Special values are defined as:
- 9999.9 Missing value

**pathAtten** (4-byte float, array size: nray x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
- 9999.9 Missing value

**reliabFactor** (4-byte float, array size: nray x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
- 9999.9 Missing value

**reliabFlag** (2-byte integer, array size: nray x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:

- 1 if Rel_eff > 3 ; PIAeff estimate is considered reliable
- 2 if 3 \( \geq \) Rel_eff > 1 ; PIAeff estimate is considered marginally reliable
- 3 if Rel_eff \( \leq \) 1 ; PIAeff is unreliable
- 4 if SNR at surface < 2dB; provides a lower bound to the path-attenuation
- 9 (no-rain case)
Special values are defined as:
-9999  Missing value

**refScanID** (2-byte integer, array size: nearFar x foreBack x nray x nscan):
The number of scan lines between the current scan and the beginning (or end) of the
along-track reference data at each angle bin. The values are computed by the equation:
Current Scan Number - Reference Scan Number. The values are positive for the Forward
estimates and negative for the Backward estimates. The Fortran indices for nearFar
foreBack are:

1,1 - Forward - Near reference  
2,1 - Forward - Far reference  
1,2 - Backward - Near reference  
2,2 - Backward - Far reference

Special values are defined as:
-9999  Missing value

**PIAhb** (4-byte float, array size: nray x nscan):
The 2-way attenuation of HB.

Values are in dB. Special values are defined as:
-9999.9  Missing value

**PIAhybrid** (4-byte float, array size: nray x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as:
-9999.9  Missing value

**zeta** (4-byte float, array size: nray x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
-9999.9  Missing value

**stddevEff** (4-byte float, array size: nsdew x nray x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.
Special values are defined as:
-9999.9  Missing value

reliabFactorHY (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9  Missing value

stddevHY (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9  Missing value

reliabFlagHY (2-byte integer, array size: nray x nscan):
TBD.

Special values are defined as:
-9999  Missing value

DSD (Group)

phase (1-byte char, array size: nbin x nray x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

phase < 100  Temperature(C)=phase-100
phase > 200  Temperature(C)=phase-200
phase = 100  Top of the bright band
phase = 200  Bottom of the bright band
phase = 125 is used for the range bins between
         the top and peak of bright band
phase = 175 is used for the range bins between
         the peak and bottom of bright band

Integer values of phase/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

binNode (2-byte integer, array size: nNode x nray x nscan):

The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to
   binRealSurface in PRE group.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
-9999 - Missing

Experimental (Group)

precipRateESurface2 (4-byte float, array size: nray x nscan):
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateESurface2Status (1-byte char, array size: nray x nscan):
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
255 Missing value

sigmaZeroProfile (4-byte float, array size: nbinSZP x nray x nscan):
Surface backscattering cross section profile around the current ifov. For information on
this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:
-9999.9 Missing value

binDEML2 (2-byte integer, array size: nray x nscan):
Range bin number of the digital elevation model surface estimate. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
-9999 Missing value

seaIceConcentration (4-byte float, array size: nray x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value

SLV (Group)

flagSLV (1-byte integer, array size: nbin x nray x nscan):
Special values are defined as:
-99 Missing value

paramDSD (4-byte float, array size: nDSD x nbin x nray x nscan):
Parameters of the drop size distribution. The first index is dBNw; the second index is Dm in mm. Special values are defined as:
-9999.9 Missing value

binEchoBottom (2-byte integer, array size: nray x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

piaFinal (4-byte float, array size: nray x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

sigmaZeroCorrected (4-byte float, array size: nray x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9 Missing value

zFactorCorrected (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9 Missing value
**zFactorCorrectedESurface** (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**paramNUBF** (4-byte float, array size: nNUBF x nray x nscan):
TBD. Special values are defined as:
-9999.9 Missing value

**precipRate** (4-byte float, array size: nbin x nray x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipWaterIntegrated** (4-byte float, array size: LS x nray x nscan):
Precipitation water vertically integrated. Values are in g/m$^2$. Special values are defined as:
-9999.9 Missing value

**qualitySLV** (4-byte integer, array size: nray x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are defined as:
-9999 Missing value

**precipRateNearSurface** (4-byte float, array size: nray x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface** (4-byte float, array size: nray x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateAve24** (4-byte float, array size: nray x nscan):
Average of precipitation rate for 2 to 4 km height. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**phaseNearSurface** (1-byte char, array size: nray x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

- phaseNearSurface < 100 Temperature(C) = phaseNearSurface - 100
- phaseNearSurface > 200 Temperature(C) = phaseNearSurface - 200
- phaseNearSurface = 100 Top of the bright band
phaseNearSurface = 200 Bottom of the bright band
phaseNearSurface = 125 is used for the range bins between
the top and peak of bright band
phaseNearSurface = 175 is used for the range bins between
the peak and bottom of bright band

Integer values of phaseNearSurface/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

epsilon (4-byte float, array size: nbin x nray x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution,
epsilon = 1 is no adjustment. Special values are defined as:
-9999.9 Missing value

FLG (Group)

flagEcho (1-byte integer, array size: nbin x nray x nscan):

Flag of precipitation and main/side lobe clutter
information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>

qualityData (4-byte integer, array size: nray x nscan):
Normal data gives “0”. Non-zero values mean the kinds of errors. Special values are
defined as:
-9999 Missing value
Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag has 2 bits of information.

The 2 bit flag for each module has values:

[higher bit  lower bit]
[0 0] Good
[0 1] Warning but usable
[1 0] NG or error

The bits of qualityData are assigned as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 7</td>
<td>Copy of dataQuality in level 1B product</td>
</tr>
<tr>
<td>8 - 9</td>
<td>Flag by input module</td>
</tr>
<tr>
<td>10 - 11</td>
<td>Flag by preparation module</td>
</tr>
<tr>
<td>12 - 13</td>
<td>Flag by vertical module</td>
</tr>
<tr>
<td>14 - 15</td>
<td>Flag by classification module</td>
</tr>
<tr>
<td>16 - 17</td>
<td>Flag by SRT module</td>
</tr>
<tr>
<td>18 - 19</td>
<td>Flag by DSD module</td>
</tr>
<tr>
<td>20 - 21</td>
<td>Flag by solver module</td>
</tr>
<tr>
<td>22 - 23</td>
<td>Flag by output module</td>
</tr>
<tr>
<td>24 - 31</td>
<td>Spare</td>
</tr>
</tbody>
</table>

**qualityFlag** (1-byte integer, array size: nray x nscan):
Flag derived from qualityData with the following values: Special values are defined as:

-99 Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

**flagSensor** (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>
flagScanPattern (2-byte integer, array size: nscan):

Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

C Structure Header file:

```c
#ifndef _TK_2AKu_H_
#define _TK_2AKu_H_

#ifndef _L2AKu_FLG_
#define _L2AKu_FLG_

typedef struct {
    signed char flagEcho[49][176];
    int qualityData[49];
    signed char qualityFlag[49];
    signed char flagSensor;
    short flagScanPattern;
} L2AKu_FLG;

#endif

#ifndef _L2AKu_SLV_
#define _L2AKu_SLV_

typedef struct {
    signed char flagSLV[49][176];
    float paramDSD[49][176][2];
    short binEchoBottom[49];
    float piaFinal[49];
    float sigmaZeroCorrected[49];
    float zFactorCorrected[49][176];
    float zFactorCorrectedESurface[49];
    float zFactorCorrectedNearSurface[49];
    float paramNUBF[49][3];
    float precipRate[49][176];
    float precipWaterIntegrated[49][2];
    int qualitySLV[49];
} L2AKu_SLV;

#endif
```


```c
float precipRateNearSurface[49];
float precipRateESurface[49];
float precipRateAve24[49];
unsigned char phaseNearSurface[49];
float epsilon[49][176];
} L2AKu_SLV;
#endif

#ifndef _L2AKu_EXPERIMENTAL_
define _L2AKu_EXPERIMENTAL_

typedef struct {
    float precipRateESurface2[49];
    unsigned char precipRateESurface2Status[49];
    float sigmaZeroProfile[49][7];
    short binDEML2[49];
    float seaIceConcentration[49];
} L2AKu_EXPERIMENTAL;
#endif

#ifndef _L2AKu_DSD_
define _L2AKu_DSD_

typedef struct {
    unsigned char phase[49][176];
    short binNode[49][5];
} L2AKu_DSD;
#endif

#ifndef _L2AKu_SRT_
define _L2AKu_SRT_

typedef struct {
    float PIAalt[49][6];
    float RFactorAlt[49][6];
    float PIAweight[49][6];
    float pathAtten[49];
    float reliabFactor[49];
    short reliabFlag[49];
    short refScanID[49][2][2];
```
float PIAhb[49];
float PIAhybrid[49];
float zeta[49];
float stddevEff[49][3];
float reliabFactorHY[49];
float stddevHY[49];
short reliabFlagHY[49];
} L2AKu_SRT;

#endif

#ifndef _L2AKu_CSF_
define _L2AKu_CSF_

typedef struct {
    int flagBB[49];
    short binBBPeak[49];
    short binBBTop[49];
    short binBBBottom[49];
    float heightBB[49];
    float widthBB[49];
    int qualityBB[49];
    int typePrecip[49];
    int qualityTypePrecip[49];
    int flagShallowRain[49];
    signed char flagHeavyIcePrecip[49];
    signed char flagAnvil[49];
} L2AKu_CSF;

#define _L2AKu_CSF_
#endif

#ifndef _L2AKu_VER_
define _L2AKu_VER_

typedef struct {
    short binZeroDeg[49];
    float attenuationNP[49][176];
    float piaNP[49][4];
    float sigmaZeroNPCorrected[49];
    float heightZeroDeg[49];
} L2AKu_VER;

#define _L2AKu_VER_
#endif
#ifndef _L2AKu_PRE_
#define _L2AKu_PRE_

typedef struct {
    float elevation[49];
    int landSurfaceType[49];
    float localZenithAngle[49];
    int flagPrecip[49];
    unsigned char flagSigmaZeroSaturation[49];
    short binRealSurface[49];
    short binStormTop[49];
    float heightStormTop[49];
    short binClutterFreeBottom[49];
    float sigmaZeroMeasured[49];
    float zFactorMeasured[49][176];
    float ellipsoidMeasured[49][176];
    float snRatioAtRealSurface[49];
    float adjustFactor[49];
    signed char snowIceCover[49];
} L2AKu_PRE;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;

#endif
typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2AKu_SCANSTATUS;

#else

#endif

#define _L2AKu_SCANSTATUS_

#endif

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#else

#endif

#define _SCANTIME_

#endif

#else

#endif

#define _L2AKu_NS_
#define _L2AKu_NS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2AKu_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2AKu_PRE PRE;
    L2AKu_VER VER;
    L2AKu_CSF CSF;
    L2AKu_SRT SRT;
    L2AKu_DSD DSD;
    L2AKu_EXPERIMENTAL Experimental;
    L2AKu_SLV SLV;
    L2AKu_FLG FLG;
} L2AKu_NS;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L2AKu_FLG/
    BYTE flagEcho(176,49)
    INTEGER*4 qualityData(49)
    BYTE qualityFlag(49)
    BYTE flagSensor
    INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2AKu_SLV/
    BYTE flagSLV(176,49)
    REAL*4 paramDSD(2,176,49)
    INTEGER*2 binEchoBottom(49)
    REAL*4 piaFinal(49)
    REAL*4 sigmaZeroCorrected(49)
    REAL*4 zFactorCorrected(176,49)
    REAL*4 zFactorCorrectedESurface(49)
    REAL*4 zFactorCorrectedNearSurface(49)
    REAL*4 paramNUBF(3,49)
    REAL*4 precipRate(176,49)
REAL*4 precipWaterIntegrated(2,49)
INTEGER*4 qualitySLV(49)
REAL*4 precipRateNearSurface(49)
REAL*4 precipRateESurface(49)
REAL*4 precipRateAve24(49)
CHARACTER phaseNearSurface(49)
REAL*4 epsilon(176,49)

END STRUCTURE

STRUCTURE /L2AKu_EXPERIMENTAL/
  REAL*4 precipRateESurface2(49)
  CHARACTER precipRateESurface2Status(49)
  REAL*4 sigmaZeroProfile(7,49)
  INTEGER*2 binDEML2(49)
  REAL*4 seaIceConcentration(49)

END STRUCTURE

STRUCTURE /L2AKu_DSD/
  CHARACTER phase(176,49)
  INTEGER*2 binNode(5,49)

END STRUCTURE

STRUCTURE /L2AKu_SRT/
  REAL*4 PIAalt(6,49)
  REAL*4 RFactorAlt(6,49)
  REAL*4 PIAweight(6,49)
  REAL*4 pathAtten(49)
  REAL*4 reliabFactor(49)
  INTEGER*2 reliabFlag(49)
  INTEGER*2 refScanID(2,2,49)
  REAL*4 PIAhb(49)
  REAL*4 PIAhybrid(49)
  REAL*4 zeta(49)
  REAL*4 stddevEff(3,49)
  REAL*4 reliabFactorHY(49)
  REAL*4 stddevHY(49)
  INTEGER*2 reliabFlagHY(49)

END STRUCTURE

STRUCTURE /L2AKu_CSF/
  INTEGER*4 flagBB(49)
  INTEGER*2 binBBPeak(49)
  INTEGER*2 binBBTop(49)
INTEGER*2 binBBBottom(49)
REAL*4 heightBB(49)
REAL*4 widthBB(49)
INTEGER*4 qualityBB(49)
INTEGER*4 typePrecip(49)
INTEGER*4 qualityTypePrecip(49)
INTEGER*4 flagShallowRain(49)
BYTE flagHeavyIcePrecip(49)
BYTE flagAnvil(49)
END STRUCTURE

STRUCTURE /L2AKu_VER/
    INTEGER*2 binZeroDeg(49)
    REAL*4 attenuationNP(176,49)
    REAL*4 piaNP(4,49)
    REAL*4 sigmaZeroNPCorrected(49)
    REAL*4 heightZeroDeg(49)
END STRUCTURE

STRUCTURE /L2AKu_PRE/
    REAL*4 elevation(49)
    INTEGER*4 landSurfaceType(49)
    REAL*4 localZenithAngle(49)
    INTEGER*4 flagPrecip(49)
    CHARACTER flagSigmaZeroSaturation(49)
    INTEGER*2 binRealSurface(49)
    INTEGER*2 binStormTop(49)
    REAL*4 heightStormTop(49)
    INTEGER*2 binClutterFreeBottom(49)
    REAL*4 sigmaZeroMeasured(49)
    REAL*4 zFactorMeasured(176,49)
    REAL*4 ellipsoidBinOffset(49)
    REAL*4 snRatioAtRealSurface(49)
    REAL*4 adjustFactor(49)
    BYTE snowIceCover(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
    REAL*4 scPos(3)
    REAL*4 scVel(3)
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L2AKu_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AKu_NS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
5.49 2AKa - Ka precipitation

The Ka Level-2A product, 2AKa, "Ka precipitation," is written as a 2 swath structure. The swaths are MS, matched scans, and HS, high sensitivity scans. The following sections describe the structure and contents of the format.

Dimension definitions:
- nscan var Number of scans in the granule.
- nrayMS 25 Number of angle bins in each MS scan.
- nrayHS 24 Number of angle bins in each HS scan.
- nbin 176 Number of range bins in each NS and MS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
- nbinHS 88 Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.
- nbinSZP 7 Number of range bins for sigmaZeroProfile.
- nbinSZPHS 5 Number of range bins for sigmaZeroProfile in each HS scan.
- nNP 4 Number of NP kinds.
- foreBack 2 Forward, Backward.
- method 6 Number of SRT methods.
- nsdew 3 Number of standard deviation effective ways.
- nNode 5 Number of binNode.
- nDSD 2 Number of DSD parameters. Parameters are dBNw and Dm (mm).
- LS 2 Liquid, solid.
- nNUBF 3 Number of NUBF parameters.

Figure 525 through Figure 549 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 525: Data Format Structure for 2AKa, Ka precipitation

Figure 526: Data Format Structure for 2AKa, MS
Figure 527: Data Format Structure for 2AKa, HS

Figure 528: Data Format Structure for 2AKa, MS, ScanTime
5.49 2AKa - Ka precipitation

Figure 529: Data Format Structure for 2AKa, MS, scanStatus

Figure 530: Data Format Structure for 2AKa, MS, navigation
**PRE**

- elevation: 4 bytes, Array: nrayMS x nscan
- landSurfaceType: 4 bytes, Array: nrayMS x nscan
- localZenithAngle: 4 bytes, Array: nrayMS x nscan
- flagPrecip: 4 bytes, Array: nrayMS x nscan
- flagSigmaZeroSaturation: 1 byte, Array: nrayMS x nscan
- binRealSurface: 2 bytes, Array: nrayMS x nscan
- binStormTop: 2 bytes, Array: nrayMS x nscan
- heightStormTop: 4 bytes, Array: nrayMS x nscan
- binClutterFreeBottom: 2 bytes, Array: nrayMS x nscan
- sigmaZeroMeasured: 4 bytes, Array: nrayMS x nscan
- zFactorMeasured: 4 bytes, Array: nbin x nrayMS x nscan
- ellipsoidBinOffset: 4 bytes, Array: nrayMS x nscan
- snRatioAtRealSurface: 4 bytes, Array: nrayMS x nscan
- adjustFactor: 4 bytes, Array: nrayMS x nscan
- snowIceCover: 1 byte, Array: nrayMS x nscan

**VER**

- binZeroDeg: 2 bytes, Array: nrayMS x nscan
- attenuationNP: 4 bytes, Array: nbin x nrayMS x nscan
- piaNP: 4 bytes, Array: nNP x nrayMS x nscan
- sigmaZeroNPCorrected: 4 bytes, Array: nrayMS x nscan
- heightZeroDeg: 4 bytes, Array: nrayMS x nscan

**Figure 531: Data Format Structure for 2AKa, MS, PRE**

**Figure 532: Data Format Structure for 2AKa, MS, VER**
Figure 536: Data Format Structure for 2AKa, MS, Experimental
Figure 537: Data Format Structure for 2AKa, MS, SLV

Figure 538: Data Format Structure for 2AKa, MS, FLG
5 STANDARD GPM PRODUCTS

Figure 539: Data Format Structure for 2AKa, HS, ScanTime

Figure 540: Data Format Structure for 2AKa, HS, scanStatus
Figure 541: Data Format Structure for 2AKa, HS, navigation
Figure 542: Data Format Structure for 2AKa, HS, PRE

Figure 543: Data Format Structure for 2AKa, HS, VER
**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**MS** (Swath)

**MS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in MS)
A UTC time associated with the scan.
Figure 548: Data Format Structure for 2AKa, HS, SLV

Figure 549: Data Format Structure for 2AKa, HS, FLG
**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

**Latitude** (4-byte float, array size: nrayMS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**Longitude** (4-byte float, array size: nrayMS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value
scanStatus (Group in MS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit =
1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other
bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^{**i}$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
</table>
Nominal pointing in Mission Science Mode
1 GPS point solution stale and PVT ephemeris used
2 GEONS solution stale and GEONS ephemeris used
-8000 Non-nominal mission science orientation
-9999 Missing

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
</tbody>
</table>
5.49  2AKa - Ka precipitation

4  Ku/Ka SSPA Analysis
5  Ku/Ka LNA Analysis
6  Ku/Ka Health-Check
7  Ku/Ka Standby VPRF Table OUT
8  Ku/Ka Standby Phase Out
9  Ku/Ka Standby Dump Out
10 Ku/Ka Standby (No Science Data)
11 Ku/Ka Independent Observation
12 Ku/Ka Independent External Calibration
13 Ku/Ka Independent Internal Calibration
14 Ku/Ka Independent SSPA Analysis
15 Ku/Ka Independent LNA Analysis
16 Ku/Ka Independent Health-Check
17 Ku/Ka Independent Standby VPRF Table OUT
18 Ku/Ka Independent Standby Phase Out
19 Ku/Ka Independent Standby Dump Out
20 Ku/Ka Independent Standby (No Science Data)

**limitErrorFlag** (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. `limitErrorFlag` may be used in `modeStatus`. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, `FractionalGranuleNumber = 10.5` means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**navigation** (Group in MS)

**scPos** (4-byte float, array size: XYZ x nscan):
The position vector(m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9  Missing value
**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector \((\text{ms}^{-1})\) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**PRE** (Group in MS)
**elevation** (4-byte float, array size: nrayMS x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:
-9999.9 Missing value

**landSurfaceType** (4-byte integer, array size: nrayMS x nscan):

Land surface type.

0 – 99 Ocean
100 – 199 Land
200 – 299 Coast
300 – 399 Inland water
-9999 Missing value

**localZenithAngle** (4-byte float, array size: nrayMS x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values are in degree. Special values are defined as:
-9999.9 Missing value

**flagPrecip** (4-byte integer, array size: nrayMS x nscan):

Precipitation or no precipitation.

For L2 Ku and L2 Ka

0 No precipitation
1 Precipitation
-9999 Missing value

For L2 DPR

0 No precipitation by both Ku and Ka
1 Precipitation by Ka, no rain by Ku
10 Precipitation by Ku, no rain by Ka
11 Precipitation by both Ku and Ka
-9999 Missing value

**flagSigmaZeroSaturation** (1-byte char, array size: nrayMS x nscan):

A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of
sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nrayMS x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nrayMS x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

heightStormTop (4-byte float, array size: nrayMS x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value

binClutterFreeBottom (2-byte integer, array size: nrayMS x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

sigmaZeroMeasured (4-byte float, array size: nrayMS x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:
-9999.9 Missing value

zFactorMeasured (4-byte float, array size: nbin x nrayMS x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:
-9999.9 Missing value

ellipsoidBinOffset (4-byte float, array size: nrayMS x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

ellipsoidBinOffset =
  scRangeEllipsoid - { startBinRange + (binEllipsoid-1) x rangeBinSize}

scRangeEllipsoid : Distance between a sensor and the ellipsoid [m]
startBinRange : Distance between a sensor and a center
of the highest observed range bin [m]

binEllipsoid : Range bin number of the Ellipsoid (1 - 260)
rangeBinSize : Range bin size [m]

-9999 Missing value

\textbf{snRatioAtRealSurface} (4-byte float, array size: nrayMS x nscan):
Signal/Noise ratio at real surface range bin.

\[ snRatioAtRealSurface = 10. \times \log_{10}(\text{echoPowertrueV[mW]}/\text{noisePowertrueV[mW]}) \]

-9999 Missing value

\textbf{adjustFactor} (4-byte float, array size: nrayMS x nscan):
Adjustment factor (dB) for \text{zFactorMeasured} (dBZm') and \text{sigmaZeroMeasured} (dBs0m').
dBZm' and dBs0m' are used and stored as follows:

\[ \text{dBZm'} = \text{dBZm} - \text{adjustFactor} \]
\[ \text{dBs0m'} = \text{dBs0m} - \text{adjustFactor} \]
The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
temporal adjustment for orbit number dependency.

\textbf{snowIceCover} (1-byte integer, array size: nrayMS x nscan):
TBD. Special values are defined as:
-99 Missing value

\textbf{VER} (Group in MS)

\textbf{binZeroDeg} (2-byte integer, array size: nrayMS x nscan):

Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

**attenuationNP** (4-byte float, array size: nbin x nrayMS x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are defined as:
-9999.9 Missing value

**piaNP** (4-byte float, array size: nNP x nrayMS x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are defined as:
-9999.9 Missing value

**sigmaZeroNPCorrected** (4-byte float, array size: nrayMS x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

**heightZeroDeg** (4-byte float, array size: nrayMS x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9 Missing value

**CSF** (Group in MS)

**flagBB** (4-byte integer, array size: nrayMS x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.

**L2 DPR:**
0 no Bright Band
1 Bright Band detected by Ku and DFRm
2 Bright Band detected by Ku only
3 Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value

**L2 Ku and L2 Ka:**
0 BB not detected
1 BB detected
-1111 No rain value
-9999 Missing value
binBBPeak (2-byte integer, array size: nrayMS x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

binBBTop (2-byte integer, array size: nrayMS x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

binBBBottom (2-byte integer, array size: nrayMS x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

heightBB (4-byte float, array size: nrayMS x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

widthBB (4-byte float, array size: nrayMS x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

qualityBB (4-byte integer, array size: nrayMS x nscan):

Quality of the bright band.
When the bright band is detected, a larger positive number indicates lower confidence in the detection.

The Ku detection is clear, but the Ka and DPR detection is somewhat doubtful.

The meaning of qualityBB has not
been finalized.

3  Smeared bright band
2  Not so clear bright band
1  Clear bright band
0  BB not detected in the case of rain
-1111  No rain value
-9999  Missing value

typePrecip (4-byte integer, array size: nrayMS x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, onvective, and other, can be obtained as follows:

When typePrecip is greater than zero,
Major rain type = typePrecip/10000000
   = 1  stratiform
   = 2  convective
   = 3  other

-1111  No rain value
-9999  Missing value

Let abcdefgh be the 8 digit number,

abedefgh

then
  a: Main rain type. (a=1,2,3),
  b: 0,
  c: 0,
  d: V rain type,
  e: H rain type,
  f: BB,
  g: Shallow rain,
  h: Small size cell.

---------------------------------------------------
The following numbers appear as Ku and Ka (MS/HS) rain types:
   ---- stratiform
      1001H100
      10031000
   ---- convective
      2001H1xy (x>0 or y>0)
      2002Hbxy
      200310xy (x>0 or y>0)
200320xy
    ---- other
300330xy
where H is the rain type by H-method, and b depends on BB,
x on shallow rain and y on small size cell:
H = 1: stratiform by H-method,
   2: convective by H-method,
   3: other by H-method.

b = 0: BB not detected,
    1: BB detected.

x = 0: No shallow rain,
    1: Shallow isolated,
    3: Shallow non-isolated.

y = 0: No small size cell,
    1: Single cell,
    2: Small size cell consisting of two adjacent pixels.

In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is
also included in typePrecip and can be obtained as follows:

DFRm rain type = (typePrecip%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000 in FORTRAN

DFRm rain type
    = 1  stratiform
    = 2  convective
    = 4  transition
    = 8  DFRm method cannot be applicable at Part B (in this case
         the conventional method determines the major rain type)
    = 9  DFRm method cannot be applicable at Part A (in this case
         the conventional method determines the major rain type)

-1111  No rain value
-9999  Missing value

If dual frequency data is not available
but Ku-only or Ka-only is available,
rain type is expressed by the following 8 digit number:
    10xxxxxx --- stratiform,
    20xxxxxx --- convective,
30xxxxxx --- other, which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by
1qxxxxxx --- stratiform,
2qxxxxxx --- convective,
3qxxxxxx --- other,
where q>0.
Thus, by examining q, users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.

For MS and HS, DFRm method is used.

DFRm decision classifies rain type into
- stratiform,
- convective,
and
- transition.

The DPR numbering rule can be summarized as follows:
Let opqrstuv be the 8 digit number, then
- o: Main rain type. (o=1,2,3),
- p: DFRm rain type. (p=0,1,2,4,8,9, with p=0 for single frequency data only),
- q: DFRm BB. (q=0,1),
- r: V rain type (by conventional V-method).
  Basically r=0 for inner swath and r>0 for outer swath.
  However, r>0 when only single frequency data is available,
- s: H rain type,
- t: = 0 for inner swath,
  1 when BB is detected in the outer swath.
- u: Shallow rain,
- v: Small size cell.

DFRm type can be obtained by examining p

The meaning of p is as follows:
p = 0: single frequency data only (dual frequency data not available),
  1: stratiform by DFRm method,
  2: convective by DFRm method,
  4: transition by DFRm method,
8: DFRm decision not available,
9: DFRm decision not available.
Note that p>0 always in DPR processing, which is different from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, p=0 always.

The following numbers appear as DPR rain types:

* For NS outer swath *

--- stratiform
1901H100
19031000
--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy
--- other
390330xy

* For NS inner swath and MS *

--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type.
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type.
--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
29010xy --- H decision only (x>0 or y>0)
29020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
(x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type

**************
* For HS *
**************
--- stratiform
11B0H000
14B01000
19001000 --- H decision only
--- convective
21B0H0x0 (x>0)
22B0H0x0
240010x0 (x>0, 24B010x0 with B=0)
240020x0
241010x0 (x>0, 24B010x0 with B=1)
290010x0 (x>0) --- H decision only
290020x0 --- H decision only
--- other
340030x0
390030x0 --- H decision only

where w depends on BB by conventional V-method, B on BB
by DFRm method, H on H-method, x on shallow rain
and y on small size cell:

w = 0: BB not detected by conventional V-method,
     1: BB detected by conventional V-method.

B = 0: BB not detected by DFRm method,
1: BB detected by DFRm method.

\[ H = \begin{cases} 
1 & \text{stratiform by H-method}, \\
2 & \text{convective by H-method}, \\
3 & \text{other by H-method}.
\end{cases} \]

\[ x = \begin{cases} 
0 & \text{No shallow rain}, \\
1 & \text{Shallow isolated}, \\
3 & \text{Shallow non-isolated}.
\end{cases} \]

\[ y = \begin{cases} 
0 & \text{No small size cell}, \\
1 & \text{Single cell}, \\
2 & \text{Small size cell consisting of two adjacent pixels}.
\end{cases} \]

In the above, \( x>0 \) and \( y>0 \) are taken care of in the function \( R\_type\_classification\_dpr2() \).

qualityTypePrecip (4-byte integer, array size: nrayMS x nscan):

Quality of the precipitation type.

1 Good
-1111 No rain value
-9999 Missing value

flagShallowRain (4-byte integer, array size: nrayMS x nscan):

Type of shallow rain
\begin{align*}
0 & \text{No shallow rain} \\
10 & \text{Shallow isolated (maybe)} \\
11 & \text{Shallow isolated (certain)} \\
20 & \text{Shallow non-isolated (maybe)} \\
21 & \text{Shallow non-isolated (certain)}
\end{align*}

-1111 No rain value
-9999 Missing value

flagHeavyIcePrecip (1-byte integer, array size: nrayMS x nscan):

This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:

-99 Missing value
SRT (Group in MS)

PIAalt (4-byte float, array size: method x nrayMS x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where
- $\text{PIA}_{\text{alt}}(j=1) = \text{PIA}_{\text{Ku}}$ from forward along-track spatial at kth angle bin
- $\text{PIA}_{\text{alt}}(j=2) = \text{PIA}_{\text{Ku}}$ from backward along-track spatial at kth angle bin
- $\text{PIA}_{\text{alt}}(j=3) = \text{PIA}_{\text{Ku}}$ from forward hybrid at kth angle bin
- $\text{PIA}_{\text{alt}}(j=4) = \text{PIA}_{\text{Ku}}$ from backward hybrid at kth angle bin
- $\text{PIA}_{\text{alt}}(j=5) = \text{PIA}_{\text{Ku}}$ from temporal reference at kth angle bin
- $\text{PIA}_{\text{alt}}(j=6) = \text{PIA}_{\text{Ku}}$ from light-rain temporal reference at kth angle bin

Values are in dB. Special values are defined as:
- -9999.9 Missing value

RFactorAlt (4-byte float, array size: method x nrayMS x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAalt. Special values are defined as:
- -9999.9 Missing value

PIAweight (4-byte float, array size: method x nrayMS x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and $\sigma_j$ is the standard deviation of reference data for method j.

$\text{PIA}_{\text{weight}}(j) = \frac{1}{\sigma_j^2} \times \left( \frac{1}{\sum_j (1/\sigma_j^2)} \right)$

Special values are defined as:
- -9999.9 Missing value

pathAtten (4-byte float, array size: nrayMS x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
- -9999.9 Missing value

reliabFactor (4-byte float, array size: nrayMS x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
- -9999.9 Missing value

reliabFlag (2-byte integer, array size: nrayMS x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:
- $= 1$ if $\text{Rel}_{\text{eff}} > 3$; PIAeff estimate is considered reliable
- $= 2$ if $3 \geq \text{Rel}_{\text{eff}} > 1$; PIAeff estimate is considered marginally reliable
- $= 3$ if $\text{Rel}_{\text{eff}} \leq 1$; PIAeff is unreliable
- $= 4$ if SNR_at surface < 2dB; provides a lower bound to the path-attenuation
- $= 9$ (no-rain case)
Special values are defined as:
-9999  Missing value

**refScanID** (2-byte integer, array size: nearFar x foreBack x nrayMS x nscan):
The number of scan lines between the current scan and the beginning (or end) of the along-track reference data at each angle bin. The values are computed by the equation: Current Scan Number - Reference Scan Number. The values are positive for the Forward estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:

1,1 - Forward - Near reference
2,1 - Forward - Far reference
1,2 - Backward - Near reference
2,2 - Backward - Far reference

Special values are defined as:
-9999  Missing value

**PIAhb** (4-byte float, array size: nrayMS x nscan):
The 2-way attenuation of HB.

Values are in dB. Special values are defined as:
-9999.9  Missing value

**PIAhybrid** (4-byte float, array size: nrayMS x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as:
-9999.9  Missing value

**zeta** (4-byte float, array size: nrayMS x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
-9999.9  Missing value

**stddevEff** (4-byte float, array size: nsdew x nrayMS x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.
Special values are defined as:
-9999.9 Missing value

**reliabFactorHY** (4-byte float, array size: nrayMS x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

**stddevHY** (4-byte float, array size: nrayMS x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

**reliabFlagHY** (2-byte integer, array size: nrayMS x nscan):
TBD.

Special values are defined as:
-9999 Missing value

**DSD** (Group in MS)

**phase** (1-byte char, array size: nbin x nrayMS x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

phase < 100 Temperature(C)=phase-100
phase > 200 Temperature(C)=phase-200
phase = 100 Top of the bright band
phase = 200 Bottom of the bright band
phase = 125 is used for the range bins between
the top and peak of bright band
phase = 175 is used for the range bins between
the peak and bottom of bright band

Integer values of phase/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

**binNode** (2-byte integer, array size: nNode x nrayMS x nscan):

The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to
   binRealSurface in PRE group.

For NS and MS swaths, bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
-9999 - Missing

**Experimental** (Group in MS)

**precipRateESurface2** (4-byte float, array size: nrayMS x nscan):
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface2Status** (1-byte char, array size: nrayMS x nscan):
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
255 Missing value

**sigmaZeroProfile** (4-byte float, array size: nbinsZP x nrayMS x nscan):
Surface backscattering cross section profile around the current ifov. For information on
this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:
-9999.9 Missing value

**binDEML2** (2-byte integer, array size: nrayMS x nscan):
Range bin number of the digital elevation model surface estimate. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
-9999 Missing value

**seaIceConcentration** (4-byte float, array size: nrayMS x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value

**SLV** (Group in MS)

**flagSLV** (1-byte integer, array size: nbin x nrayMS x nscan):
Special values are defined as:
-99 Missing value

**paramDSD** (4-byte float, array size: nDSD x nbin x nrayMS x nscan):
Parameters of the drop size distribution. The first index is dBNw; the second index is Dm in mm. Special values are defined as:
-9999.9 Missing value

**binEchoBottom** (2-byte integer, array size: nrayMS x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

**piaFinal** (4-byte float, array size: nrayMS x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

**sigmaZeroCorrected** (4-byte float, array size: nrayMS x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9 Missing value

**zFactorCorrected** (4-byte float, array size: nbin x nrayMS x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9 Missing value
**zFactorCorrectedESurface** (4-byte float, array size: nrayMS x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nrayMS x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**paramNUBF** (4-byte float, array size: nNUBF x nrayMS x nscan):
TBD. Special values are defined as:
-9999.9 Missing value

**precipRate** (4-byte float, array size: nbin x nrayMS x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipWaterIntegrated** (4-byte float, array size: LS x nrayMS x nscan):
Precipitation water vertically integrated. Values are in g/m². Special values are defined as:
-9999.9 Missing value

**qualitySLV** (4-byte integer, array size: nrayMS x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are defined as:
-9999 Missing value

**precipRateNearSurface** (4-byte float, array size: nrayMS x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface** (4-byte float, array size: nrayMS x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateAve24** (4-byte float, array size: nrayMS x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**phaseNearSurface** (1-byte char, array size: nrayMS x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

phaseNearSurface < 100 Temperature(C)=phaseNearSurface-100
phaseNearSurface > 200 Temperature(C)=phaseNearSurface-200
phaseNearSurface = 100 Top of the bright band
phaseNearSurface = 200 Bottom of the bright band
phaseNearSurface = 125 is used for the range bins between
the top and peak of bright band
phaseNearSurface = 175 is used for the range bins between
the peak and bottom of bright band

Integer values of phaseNearSurface/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

epsilon (4-byte float, array size: nbin x nrayMS x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution,
epsilon = 1 is no adjustment. Special values are defined as:
  -9999.9 Missing value

FLG (Group in MS)

flagEcho (1-byte integer, array size: nbin x nrayMS x nscan):

Flag of precipitation and main/side lobe clutter
information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>

qualityData (4-byte integer, array size: nrayMS x nscan):
Normal data gives "0". Non-zero values mean the kinds of errors. Special values are
defined as:
  -9999 Missing value
Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag has 2 bits of information.

The 2 bit flag for each module has values:

[higher bit lower bit]
[0 0] Good
[0 1] Warning but usable
[1 0] NG or error

The bits of qualityData are assigned as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 7</td>
<td>Copy of dataQuality in level 1B product</td>
</tr>
<tr>
<td>8 - 9</td>
<td>Flag by input module</td>
</tr>
<tr>
<td>10 - 11</td>
<td>Flag by preparation module</td>
</tr>
<tr>
<td>12 - 13</td>
<td>Flag by vertical module</td>
</tr>
<tr>
<td>14 - 15</td>
<td>Flag by classification module</td>
</tr>
<tr>
<td>16 - 17</td>
<td>Flag by SRT module</td>
</tr>
<tr>
<td>18 - 19</td>
<td>Flag by DSD module</td>
</tr>
<tr>
<td>20 - 21</td>
<td>Flag by solver module</td>
</tr>
<tr>
<td>22 - 23</td>
<td>Flag by output module</td>
</tr>
<tr>
<td>24 - 31</td>
<td>Spare</td>
</tr>
</tbody>
</table>

**qualityFlag** (1-byte integer, array size: nrayMS x nscan):
Flag derived from qualityData with the following values: Special values are defined as:
-99  Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

**flagSensor** (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>
flagScanPattern (2-byte integer, array size: nscan):

Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

HS (Swath)

HS_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in HS)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value
**Millisecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: nrayHS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nrayHS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in HS)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
</tbody>
</table>
Operational mode is not observation mode
GPS status is abnormal
Spare (always 0)
Check sum of L1A is abnormal

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

- Bit Meaning if bit = 1
  - 0 Scan is missing
  - 1 Science telemetry packet missing
  - 2 Science telemetry segment within packet missing
  - 3 Science telemetry other missing
  - 4 Housekeeping (HK) telemetry packet missing
  - 5 Spare (always 0)
  - 6 Spare (always 0)
  - 7 Spare (always 0)

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

- Bit Meaning if bit = 1
  - 0 Spare (always 0)
  - 1 SCorientation not 0 or 180
  - 2 pointingStatus not 0
  - 3 Non-routine limitErrorFlag
  - 4 Non-routine operationalMode (not 1 or 11)
  - 5 Spare (always 0)
  - 6 Spare (always 0)
  - 7 Spare (always 0)

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0 Latitude limit exceeded for viewed pixel locations
1 Negative scan time, invalid input
2 Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in
dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be
useful. A zero integer value indicates usual geolocation. A non-zero value broken down
into the following bit flags indicates the following, where bit 0 is the least significant bit
(i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

Bit Meaning if bit = 1
0 Ephemeris Gap Interpolated
1 Attitude Gap Interpolated
2 Attitude jump/discontinuity
3 Attitude out of range
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
SCOrientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCOrientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>
**targetSelectionMidScan** (1-byte integer, array size: nscan):

targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

**limitErrorFlag** (1-byte integer, array size: nscan):
Bit flags for every ray with information about echo power limit checks.  
limitErrorFlag may be used in modeStatus.  
Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):  
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:  
-9999.9 Missing value

**navigation** (Group in HS)

**scPos** (4-byte float, array size: XYZ x nscan):  
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:  
-9999.9 Missing value

**scVel** (4-byte float, array size: XYZ x nscan):  
The velocity vector (ms$^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:  
-9999.9 Missing value

**scLat** (4-byte float, array size: nscan):  
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:  
-9999.9 Missing value

**scLon** (4-byte float, array size: nscan):  
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:  
-9999.9 Missing value

**scAlt** (4-byte float, array size: nscan):  
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:  
-9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan):  
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000
to 500000 m. Special values are defined as:
-9999.9 Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values
range from -135 to 225 degrees. Special values are defined as:
- 9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
- 9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
- 9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
- 9999.9 Missing value

**PRE (Group in HS)**

**elevation** (4-byte float, array size: nrayHS x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:
- 9999.9 Missing value

**landSurfaceType** (4-byte integer, array size: nrayHS x nscan):

Land surface type.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 99</td>
<td>Ocean</td>
</tr>
<tr>
<td>100 - 199</td>
<td>Land</td>
</tr>
<tr>
<td>200 - 299</td>
<td>Coast</td>
</tr>
<tr>
<td>300 - 399</td>
<td>Inland water</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

**localZenithAngle** (4-byte float, array size: nrayHS x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values are in degree. Special values are defined as:
- 9999.9 Missing value

**flagPrecip** (4-byte integer, array size: nrayHS x nscan):
Precipitation or no precipitation.

For L2 Ku and L2 Ka

0   No precipitation
1   Precipitation
-9999 Missing value

For L2 DPR

0   No precipitation by both Ku and Ka
1   Precipitation by Ka, no rain by Ku
10  Precipitation by Ku, no rain by Ka
11  Precipitation by both Ku and Ka
-9999 Missing value

**flagSigmaZeroSaturation** (1-byte char, array size: nrayHS x nscan):

A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

**binRealSurface** (2-byte integer, array size: nrayHS x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

**binStormTop** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

**heightStormTop** (4-byte float, array size: nrayHS x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value
**binClutterFreeBottom** (2-byte integer, array size: nrayHS x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

**sigmaZeroMeasured** (4-byte float, array size: nrayHS x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:
-9999.9 Missing value

**zFactorMeasured** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**ellipseBinOffset** (4-byte float, array size: nrayHS x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

\[
\text{ellipseBinOffset} = \text{scRangeEllipsoid} - \{ \text{startBinRange} + (\text{binEllipsoid}-1) \times \text{rangeBinSize} \}
\]

-9999 Missing value

**snRatioAtRealSurface** (4-byte float, array size: nrayHS x nscan):
Signal/Noise ratio at real surface range bin.

\[
\text{snRatioAtRealSurface} = 10. \times \log_{10}(\text{echoPowertrueV}[\text{mW}]/\text{noisePowertrueV}[\text{mW}])
\]
-9999 Missing value

**adjustFactor** (4-byte float, array size: nrayHS x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm’) and sigmaZeroMeasured (dBs0m’). dBZm’ and dBs0m’ are used and stored as follows:

\[
\text{dBZm’} = \text{dBZm} - \text{adjustFactor} \\
\text{dBs0m’} = \text{dBs0m} - \text{adjustFactor}
\]
The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
 temporal adjustment for orbit number dependency.
snowIceCover (1-byte integer, array size: nrayHS x nscan):
TBD. Special values are defined as:
-99  Missing value

VER (Group in HS)

binZeroDeg (2-byte integer, array size: nrayHS x nscan):

Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

attenuationNP (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud
ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are
defined as:
-9999.9  Missing value

piaNP (4-byte float, array size: nNP x nrayHS x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water,
cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are
defined as:
-9999.9  Missing value

sigmaZeroNPCorrected (4-byte float, array size: nrayHS x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation
particles. Values are in dB. Special values are defined as:
-9999.9  Missing value

heightZeroDeg (4-byte float, array size: nrayHS x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9  Missing value
CSF (Group in HS)

flagBB (4-byte integer, array size: nrayHS x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.

L2 DPR:
0     no Bright Band
1     Bright Band detected by Ku and DFRm
2     Bright Band detected by Ku only
3     Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value

L2 Ku and L2 Ka:
0     BB not detected
1     BB detected
-1111 No rain value
-9999 Missing value

binBBPeak (2-byte integer, array size: nrayHS x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

binBBTop (2-byte integer, array size: nrayHS x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

binBBBottom (2-byte integer, array size: nrayHS x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

heightBB (4-byte float, array size: nrayHS x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m.
Special values are defined as:
-9999.9  Missing value

**widthBB** (4-byte float, array size: nrayHS x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m.
Special values are defined as:
-9999.9  Missing value

**qualityBB** (4-byte integer, array size: nrayHS x nscan):

Quality of the bright band.
When the bright band is detected,
a larger positive number indicates lower
certainty in the detection.

The Ku detection is clear, but
the Ka and DPR detection is
somewhat doubtful.

The meaning of qualityBB has not
been finalized.

3  Smeared bright band
2  Not so clear bright band
1  Clear bright band
0  BB not detected in the case of rain
-1111 No rain value
-9999 Missing value

**typePrecip** (4-byte integer, array size: nrayHS x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories,
stratiform, convective, and other, can be obtained as follows:

When typePrecip is greater than zero,
Major rain type = typePrecip/10000000
= 1  stratiform
= 2  convective
= 3  other

-1111 No rain value
-9999 Missing value

Let abcdefgh be the 8 digit number,
abcdefgh

then

a: Main rain type. \((a=1, 2, 3)\),
b: 0,
c: 0,
d: V rain type,
e: H rain type,
f: BB,
g: Shallow rain,
h: Small size cell.

The following numbers appear as Ku and Ka (MS/HS) rain types:

---- stratiform
1001H100
10031000

---- convective
2001H1xy \((\text{x}>0 \text{ or } \text{y}>0)\)
2002Hbxy
200310xy \((\text{x}>0 \text{ or } \text{y}>0)\)
200320xy

---- other
300330xy

where \(H\) is the rain type by H-method, and \(b\) depends on BB,
\(x\) on shallow rain and \(y\) on small size cell:

\(H = 1: \) stratiform by H-method,
\(2: \) convective by H-method,
\(3: \) other by H-method.

\(b = 0: \) BB not detected,
\(1: \) BB detected.

\(x = 0: \) No shallow rain,
\(1: \) Shallow isolated,
\(3: \) Shallow non-isolated.

\(y = 0: \) No small size cell,
\(1: \) Single cell,
\(2: \) Small size cell consisting of two adjacent pixels.

In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:
DFRm rain type = (typePrecip%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000 in FORTRAN

DFRm rain type
   = 1  stratiform
   = 2  convective
   = 4  transition
   = 8  DFRm method cannot be applicable at Part B (in this case the conventional method determines the major rain type)
   = 9  DFRm method cannot be applicable at Part A (in this case the conventional method determines the major rain type)

-1111  No rain value
-9999  Missing value

If dual frequency data is not available but Ku-only or Ka-only is available, rain type is expressed by the following 8 digit number:
   10xxxxxx --- stratiform,
   20xxxxxx --- convective,
   30xxxxxx --- other,
which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by
   1qxxxxxx --- stratiform,
   2qxxxxxx --- convective,
   3qxxxxxx --- other,
where q>0.
Thus, by examining q, users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.

For MS and HS, DFRm method is used.

DFRm decision classifies rain type into stratiform, convective, and transition.

The DPR numbering rule can be summarized as follows:
Let opqrstuv be the 8 digit number, then
o: Main rain type. (o=1,2,3),
p: DFRm rain type. (p=0,1,2,4,8,9, with p=0 for single frequency data only),
q: DFRm BB. (q=0,1),
r: V rain type (by conventional V-method).
   Basically r=0 for inner swath and r>0 for outer swath.
   However, r>0 when only single frequency data is available,
s: H rain type,
t: = 0 for inner swath,
   1 when BB is detected in the outer swath.
u: Shallow rain,
v: Small size cell.

DFRm type can be obtained by examining p

The meaning of p is as follows:
p = 0: single frequency data only (dual frequency data not available),
   1: stratiform by DFRm method,
   2: convective by DFRm method,
   4: transition by DFRm method,
   8: DFRm decision not available,
   9: DFRm decision not available.

Note that p>0 always in DPR processing, which is different from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, p=0 always.

The following numbers appear as DPR rain types:

--- stratiform
1901H100
19031000

--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy

--- other
390330xy
* For NS inner swath and MS *

--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
   or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
   or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
   or NS rain >0 but no MS rain; NS V and H determine rain type

--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
   or NS rain >0 but no MS rain; NS V and H determine rain type
   (x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
   or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
   (x>0 or y>0)
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
   or NS rain >0 but no MS rain; NS V and H determine rain type

--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
   or NS rain >0 but no MS rain; NS V and H determine rain type

* For HS *

--- stratiform
11B0H000
14B01000
where \( w \) depends on \( BB \) by conventional V-method, \( B \) on \( BB \) by DFRm method, \( H \) on H-method, \( x \) on shallow rain and \( y \) on small size cell:

- \( w = 0 \): \( BB \) not detected by conventional V-method,
  - 1: \( BB \) detected by conventional V-method.

- \( B = 0 \): \( BB \) not detected by DFRm method,
  - 1: \( BB \) detected by DFRm method.

- \( H = 1 \): stratiform by H-method,
  - 2: convective by H-method,
  - 3: other by H-method.

- \( x = 0 \): No shallow rain,
  - 1: Shallow isolated,
  - 3: Shallow non-isolated.

- \( y = 0 \): No small size cell,
  - 1: Single cell,
  - 2: Small size cell consisting of two adjacent pixels.

In the above, \( x>0 \) and \( y>0 \) are taken care of in the function `R\_type\_classification\_dpr2()`.

---

**qualityTypePrecip** (4-byte integer, array size: `nrayHS x nscan`):

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Good</td>
</tr>
<tr>
<td>-1111</td>
<td>No rain value</td>
</tr>
</tbody>
</table>
-9999 Missing value

**flagShallowRain** (4-byte integer, array size: nrayHS x nscan):

Type of shallow rain
0 No shallow rain
10 Shallow isolated (maybe)
11 Shallow isolated (certain)
20 Shallow non-isolated (maybe)
21 Shallow non-isolated (certain)
-1111 No rain value
-9999 Missing value

**flagHeavyIcePrecip** (1-byte integer, array size: nrayHS x nscan):
This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:
-99 Missing value

**SRT** (Group in HS)

**PIAalt** (4-byte float, array size: method x nrayHS x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where

\[ PIAalt (j=1) = PIA_{Ku} \text{ from forward along-track spatial at kth angle bin} \]
\[ PIAalt (j=2) = PIA_{Ku} \text{ from backward along-track spatial at kth angle bin} \]
\[ PIAalt (j=3) = PIA_{Ku} \text{ from forward hybrid at kth angle bin} \]
\[ PIAalt (j=4) = PIA_{Ku} \text{ from backward hybrid at kth angle bin} \]
\[ PIAalt (j=5) = PIA_{Ku} \text{ from temporal reference at kth angle bin} \]
\[ PIAalt (j=6) = PIA_{Ku} \text{ from light-rain temporal reference at kth angle bin} \]

Values are in dB. Special values are defined as:
-9999.9 Missing value

**RFactorAlt** (4-byte float, array size: method x nrayHS x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAalt. Special values are defined as:
-9999.9 Missing value

**PIAweight** (4-byte float, array size: method x nrayHS x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

\[ PIAweight_j = 1/\sigma_j^2 \times (1/\sum_j(1/\sigma_j^2)) \]
Special values are defined as:
-9999.9  Missing value

**pathAtten** (4-byte float, array size: nrayHS x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are
defined as:
-9999.9  Missing value

**reliabFactor** (4-byte float, array size: nrayHS x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined
as:
-9999.9  Missing value

**reliabFlag** (2-byte integer, array size: nrayHS x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability
factor (Rel_eff) in reliabFactor. Reliability Flag is:
= 1  if Rel_eff > 3 ; PIAeff estimate is considered reliable
= 2  if 3 ≥ Rel_eff > 1 ; PIAeff estimate is considered marginally reliable
= 3  if Rel_eff ≤ 1 ; PIAeff is unreliable
= 4  if SNR_at surface < 2dB; provides a lower bound to the path-attenuation
= 9  (no-rain case)

Special values are defined as:
-9999  Missing value

**refScanID** (2-byte integer, array size: nearFar x foreBack x nrayHS x nscan):
The number of scan lines between the current scan and the beginning (or end) of the
along-track reference data at each angle bin. The values are computed by the equation:
Current Scan Number - Reference Scan Number. The values are positive for the Forward
estimates and negative for the Backward estimates. The Fortran indices for nearFar
foreBack are:

1,1 - Forward - Near reference
2,1 - Forward - Far reference
1,2 - Backward - Near reference
2,2 - Backward - Far reference

Special values are defined as:
-9999  Missing value

**PIAhb** (4-byte float, array size: nrayHS x nscan):
The 2-way attenuation of HB.

Values are in dB. Special values are defined as:
-9999.9  Missing value
PIAhybrid (4-byte float, array size: nrayHS x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as:
-9999.9   Missing value

zeta (4-byte float, array size: nrayHS x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
-9999.9   Missing value

stddevEff (4-byte float, array size: nsdew x nrayHS x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.

Special values are defined as:
-9999.9   Missing value

reliabFactorHY (4-byte float, array size: nrayHS x nscan):
TBD.

Special values are defined as:
-9999.9   Missing value

stddevHY (4-byte float, array size: nrayHS x nscan):
TBD.

Special values are defined as:
-9999.9   Missing value

reliabFlagHY (2-byte integer, array size: nrayHS x nscan):
TBD.
Special values are defined as:
-9999 Missing value

**DSD (Group in HS)**

**phase** (1-byte char, array size: nbinHS x nrayHS x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

- \( \text{phase} < 100 \) \( \Rightarrow \text{Temperature(C) = phase - 100} \)
- \( \text{phase} > 200 \) \( \Rightarrow \text{Temperature(C) = phase - 200} \)
- \( \text{phase} = 100 \) Top of the bright band
- \( \text{phase} = 200 \) Bottom of the bright band
- \( \text{phase} = 125 \) is used for the range bins between
  - the top and peak of bright band
- \( \text{phase} = 175 \) is used for the range bins between
  - the peak and bottom of bright band

Integer values of \( \text{phase}/100 = \)

- 0 - solid
- 1 - mixed phase
- 2 - liquid
- 255 - Missing

**binNode** (2-byte integer, array size: nNode x nrayHS x nscan):

The bin number of the 5 nodes defined as:

- 0 - Bin number of storm top.
- 1 - Stratiform: 500m above center of bright band.
  Convective: 750m above 0deg C level.
- 2 - Stratiform: center of bright band.
  Convective: 0deg C level.
- 3 - Stratiform: 500m below center of bright band.
  Convective: 750m below 0deg C level.
- 4 - Bin number of real surface equal to
  binRealSurface in PRE group.

For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths, 
bin numbers are 1-based ranging 
from 1 at the top of the data window 
with 88 at the Ellipsoid. 
-9999 - Missing

**Experimental (Group in HS)**

precipRateESurface2 (4-byte float, array size: nrayHS x nscan): 
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as: 
-9999.9 Missing value

precipRateESurface2Status (1-byte char, array size: nrayHS x nscan): 
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as: 
255 Missing value

sigmaZeroProfile (4-byte float, array size: nbinSZPHS x nrayHS x nscan): 
Surface backscattering cross section profile around the current ifov. For information on this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as: 
-9999.9 Missing value

binDEML2 (2-byte integer, array size: nrayHS x nscan): 
Range bin number of the digital elevation model surface estimate. For information on this experimental field contact the Joint DPR Team. Special values are defined as: 
-9999 Missing value

seaIceConcentration (4-byte float, array size: nrayHS x nscan): 
Sea ice concentration estimated by Ku. For information on this experimental field contact the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as: 
-9999.9 Missing value

**SLV (Group in HS)**

flagSLV (1-byte integer, array size: nbinHS x nrayHS x nscan): 
Special values are defined as: 
-99 Missing value

paramDSD (4-byte float, array size: nDSD x nbinHS x nrayHS x nscan): 
Parameters of the drop size distribution. The first index is dB/Nw; the second index is
Dm in mm. Special values are defined as:
-9999.9  Missing value

**binEchoBottom** (2-byte integer, array size: nrayHS x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999  Missing value

**piaFinal** (4-byte float, array size: nrayHS x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9  Missing value

**sigmaZeroCorrected** (4-byte float, array size: nrayHS x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9  Missing value

**zFactorCorrected** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9  Missing value

**zFactorCorrectedESurface** (4-byte float, array size: nrayHS x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9  Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nrayHS x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9  Missing value

**paramNUBF** (4-byte float, array size: nNUBF x nrayHS x nscan):
TBD. Special values are defined as:
-9999.9  Missing value

**precipRate** (4-byte float, array size: nbinHS x nrayHS x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9  Missing value

**precipWaterIntegrated** (4-byte float, array size: LS x nrayHS x nscan):
Precipitation water vertically integrated. Values are in g/m². Special values are defined as:
-9999.9  Missing value

**qualitySLV** (4-byte integer, array size: nrayHS x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are
defined as:

-9999   Missing value

**precipRateNearSurface** (4-byte float, array size: nrayHS x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
-9999.9   Missing value

**precipRateESurface** (4-byte float, array size: nrayHS x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
-9999.9   Missing value

**precipRateAve24** (4-byte float, array size: nrayHS x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
-9999.9   Missing value

**phaseNearSurface** (1-byte char, array size: nrayHS x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

phaseNearSurface < 100   Temperature(C)=phaseNearSurface-100  
phaseNearSurface > 200   Temperature(C)=phaseNearSurface-200  
phaseNearSurface = 100   Top of the bright band  
phaseNearSurface = 200   Bottom of the bright band  
phaseNearSurface = 125   is used for the range bins between  
                         the top and peak of bright band  
phaseNearSurface = 175   is used for the range bins between  
                         the peak and bottom of bright band

Integer values of phaseNearSurface/100 =

0   - solid  
1   - mixed phase  
2   - liquid  
255   - Missing

**epsilon** (4-byte float, array size: nbinHS x nrayHS x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution, epsilon = 1 is no adjustment. Special values are defined as:
-9999.9   Missing value

**FLG** (Group in HS)
flagEcho (1-byte integer, array size: nbinHS x nrayHS x nscan):

Flag of precipitation and main/side lobe clutter information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>

qualityData (4-byte integer, array size: nrayHS x nscan):
Normal data gives ”0”. Non-zero values mean the kinds of errors. Special values are defined as:
-9999  Missing value

Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag has 2 bits of information.

The 2 bit flag for each module has values:
[higher bit lower bit]
[0 0]  Good
[0 1]  Warning but usable
[1 0]  NG or error

The bits of qualityData are assigned as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 7</td>
<td>Copy of dataQuality in level 1B product</td>
</tr>
<tr>
<td>8 - 9</td>
<td>Flag by input module</td>
</tr>
<tr>
<td>10 - 11</td>
<td>Flag by preparation module</td>
</tr>
<tr>
<td>12 - 13</td>
<td>Flag by vertical module</td>
</tr>
<tr>
<td>14 - 15</td>
<td>Flag by classification module</td>
</tr>
<tr>
<td>16 - 17</td>
<td>Flag by SRT module</td>
</tr>
<tr>
<td>18 - 19</td>
<td>Flag by DSD module</td>
</tr>
<tr>
<td>20 - 21</td>
<td>Flag by solver module</td>
</tr>
</tbody>
</table>
22 - 23 Flag by output module
24 - 31 Spare

**qualityFlag** (1-byte integer, array size: nrayHS x nscan):
Flag derived from qualityData with the following values: Special values are defined as:
-99  Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

**flagSensor** (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>

**flagScanPattern** (2-byte integer, array size: nscan):
Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**C Structure Header file:**

```c
#ifndef _TK_2AKa_H_
#define _TK_2AKa_H_

#ifndef _L2AKa_HS_FLG_
#define _L2AKa_HS_FLG_

typedef struct {
    signed char flagEcho[24][88];
    int qualityData[24];
    signed char qualityFlag[24];
}
```
signed char flagSensor;
short flagScanPattern;
} L2AKa_HS_FLG;

#endif

#ifndef _L2AKa_HS_SLV_
define _L2AKa_HS_SLV_

typedef struct {
signed char flagSLV[24][88];
float paramDSD[24][88][2];
short binEchoBottom[24];
float piaFinal[24];
float sigmaZeroCorrected[24];
float zFactorCorrected[24][88];
float zFactorCorrectedESurface[24];
float zFactorCorrectedNearSurface[24];
float paramNUBF[24][3];
float precipRate[24][88];
float precipWaterIntegrated[24][2];
int qualitySLV[24];
float precipRateNearSurface[24];
float precipRateESurface[24];
float precipRateAve24[24];
unsigned char phaseNearSurface[24];
float epsilon[24][88];
} L2AKa_HS_SLV;
#endif

#ifndef _L2AKa_HS_EXPERIMENTAL_
define _L2AKa_HS_EXPERIMENTAL_

typedef struct {
float precipRateESurface2[24];
unsigned char precipRateESurface2Status[24];
float sigmaZeroProfile[24][5];
short binDEML2[24];
float seaIceConcentration[24];
} L2AKa_HS_EXPERIMENTAL;
#endif
#ifndef _L2AKa_HS_DSD_
define _L2AKa_HS_DSD_

typedef struct {  
    unsigned char phase[24][88];
    short binNode[24][5];
} L2AKa_HS_DSD;
#endif

#endif

#endif

#define _L2AKa_HS_SRT_

typedef struct {  
    float PIAalt[24][6];
    float RFactorAlt[24][6];
    float PIAweight[24][6];
    float pathAtten[24];
    float reliabFactor[24];
    short reliabFlag[24];
    short refScanID[24][2][2];
    float PIAhb[24];
    float PIAhybrid[24];
    float zeta[24];
    float stddevEff[24][3];
    float reliabFactorHY[24];
    float stddevHY[24];
    short reliabFlagHY[24];
} L2AKa_HS_SRT;
#endif

#endif

#define _L2AKa_HS_CSF_

typedef struct {  
    int flagBB[24];
    short binBBPeak[24];
    short binBBTop[24];
    short binBBBottom[24];
    float heightBB[24];
    float widthBB[24];
int qualityBB[24];
int typePrecip[24];
int qualityTypePrecip[24];
int flagShallowRain[24];
signed char flagHeavyIcePrecip[24];
} L2AKa_HS_CSF;
#endif

#ifndef _L2AKa_HS_VER_
#define _L2AKa_HS_VER_

typedef struct {
    short binZeroDeg[24];
    float attenuationNP[24][88];
    float piaNP[24][4];
    float sigmaZeroNPCorrected[24];
    float heightZeroDeg[24];
} L2AKa_HS_VER;
#endif

#ifndef _L2AKa_HS_PRE_
#define _L2AKa_HS_PRE_

typedef struct {
    float elevation[24];
    int landSurfaceType[24];
    float localZenithAngle[24];
    int flagPrecip[24];
    unsigned char flagSigmaZeroSaturation[24];
    short binRealSurface[24];
    short binStormTop[24];
    float heightStormTop[24];
    short binClutterFreeBottom[24];
    float sigmaZeroMeasured[24];
    float zFactorMeasured[24][88];
    float ellipsoidBinOffset[24];
    float snRatioAtRealSurface[24];
    float adjustFactor[24];
    signed char snowIceCover[24];
} L2AKa_HS_PRE;
typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2AKa_HS_SCANSTATUS;

#endif

ifndef _L2AKa_HS_SCANSTATUS_
define _L2AKa_HS_SCANSTATUS_
#endif

typedef struct {
    SCANTIME ScanTime;
    float Latitude[24];
    float Longitude[24];
    L2AKa_HS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2AKa_HS_PRE PRE;
    L2AKa_HS_VER VER;
    L2AKa_HS_CSF CSF;
    L2AKa_HS_SRT SRT;
    L2AKa_HS_DSD DSD;
    L2AKa_HS_EXPERIMENTAL Experimental;
    L2AKa_HS_SLV SLV;
    L2AKa_HS_FLG FLG;
} L2AKa_HS;

#endif
5.49  2AKa - Ka precipitation

```c
#ifndef _L2AKa_MS_FLG_
define _L2AKa_MS_FLG_

typedef struct {
    signed char flagEcho[25][176];
    int qualityData[25];
    signed char qualityFlag[25];
    signed char flagSensor;
    short flagScanPattern;
} L2AKa_MS_FLG;
#endif

#ifndef _L2AKa_MS_SLV_
define _L2AKa_MS_SLV_

typedef struct {
    signed char flagSLV[25][176];
    float paramDSD[25][176][2];
    short binEchoBottom[25];
    float piaFinal[25];
    float sigmaZeroCorrected[25];
    float zFactorCorrected[25][176];
    float zFactorCorrectedESurface[25];
    float zFactorCorrectedNearSurface[25];
    float paramNUBF[25][3];
    float precipRate[25][176];
    float precipRateNearSurface[25];
    float precipRateESurface[25];
    float precipRateAve24[25];
    unsigned char phaseNearSurface[25];
    float epsilon[25][176];
} L2AKa_MS_SLV;
#endif

#ifndef _L2AKa_MS_EXPERIMENTAL_
define _L2AKa_MS_EXPERIMENTAL_

typedef struct {
```
float precipRateESurface2[25];
unsigned char precipRateESurface2Status[25];
float sigmaZeroProfile[25][7];
short binDEML2[25];
float seaIceConcentration[25];
}
L2AKa_MS_EXPERIMENTAL;
#endif

#ifndef _L2AKa_MS_DSD_
#define _L2AKa_MS_DSD_

typedef struct {
    unsigned char phase[25][176];
    short binNode[25][5];
} L2AKa_MS_DSD;
#endif

#ifndef _L2AKa_MS_SRT_
#define _L2AKa_MS_SRT_

typedef struct {
    float PIAalt[25][6];
    float RFactorAlt[25][6];
    float PIAweight[25][6];
    float pathAtten[25];
    float reliabFactor[25];
    short reliabFlag[25];
    short refScanID[25][2][2];
    float PIAhb[25];
    float PIAhybrid[25];
    float zeta[25];
    float stddevEff[25][3];
    float reliabFactorHY[25];
    float stddevHY[25];
    short reliabFlagHY[25];
} L2AKa_MS_SRT;
#endif

#ifndef _L2AKa_MS_CSF_
#define _L2AKa_MS_CSF_
typedef struct {
    int flagBB[25];
    short binBBPeak[25];
    short binBBTop[25];
    short binBBBottom[25];
    float heightBB[25];
    float widthBB[25];
    int qualityBB[25];
    int typePrecip[25];
    int qualityTypePrecip[25];
    int flagShallowRain[25];
    signed char flagHeavyIcePrecip[25];
} L2AKa_MS_CSF;

#endif

#ifndef _L2AKa_MS_VER_
define _L2AKa_MS_VER_

typedef struct {
    short binZeroDeg[25];
    float attenuationNP[25][176];
    float piaNP[25][4];
    float sigmaZeroNPCorrected[25];
    float heightZeroDeg[25];
} L2AKa_MS_VER;

#endif

#ifndef _L2AKa_MS_PRE_
define _L2AKa_MS_PRE_

typedef struct {
    float elevation[25];
    int landSurfaceType[25];
    float localZenithAngle[25];
    int flagPrecip[25];
    unsigned char flagSigmaZeroSaturation[25];
    short binRealSurface[25];
    short binStormTop[25];
    float heightStormTop[25];
    short binClutterFreeBottom[25];
} L2AKa_MS_PRE;

#endif
float sigmaZeroMeasured[25];
float zFactorMeasured[25][176];
float ellipsoidBinOffset[25];
float snRatioAtRealSurface[25];
float adjustFactor[25];
signed char snowIceCover[25];

} L2AKa_MS_PRE;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
  float scPos[3];
  float scVel[3];
  float scLat;
  float scLon;
  float scAlt;
  float dprAlt;
  float scAttRollGeoc;
  float scAttPitchGeoc;
  float scAttYawGeoc;
  float scAttRollGeod;
  float scAttPitchGeod;
  float scAttYawGeod;
  float greenHourAng;
  double timeMidScan;
  double timeMidScanOffset;
} NAVIGATION;

#endif

#ifndef _L2AKa_MS_SCANSTATUS_
#define _L2AKa_MS_SCANSTATUS_

typedef struct {
  signed char dataQuality;
  signed char dataWarning;
  signed char missing;
  signed char modeStatus;
  short geoError;
  short geoWarning;

  } L2AKa_MS_SCANSTATUS;

#endif
typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

typedef struct {
    SCANTIME ScanTime;
    float Latitude[25];
    float Longitude[25];
    L2AKa_MS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2AKa_MS_PRE PRE;
    L2AKa_MS_VER VER;
    L2AKa_MS_CSF CSF;
    L2AKa_MS_SRT SRT;
    L2AKa_MS_DSD DSD;
    L2AKa_MS_EXPERIMENTAL Experimental;
} L2AKa_MS_SCANSTATUS;
L2AKa_MS_SLV SLV;
L2AKa_MS_FLG FLG;
} L2AKa_MS;

#endif

#ifndef _L2AKa_SWATHS_
define _L2AKa_SWATHS_

typedef struct {
    L2AKa_MS MS;
    L2AKa_HS HS;
} L2AKa_SWATHS;

#endif

#define _L2AKa_SWATHS_

Fortran Structure Header file:

STRUCTURE /L2AKa_HS_FLG/
    BYTE flagEcho(88,24)
    INTEGER*4 qualityData(24)
    BYTE qualityFlag(24)
    BYTE flagSensor
    INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2AKa_HS_SLV/
    BYTE flagSLV(88,24)
    REAL*4 paramDSD(2,88,24)
    INTEGER*2 binEchoBottom(24)
    REAL*4 piaFinal(24)
    REAL*4 sigmaZeroCorrected(24)
    REAL*4 zFactorCorrected(88,24)
    REAL*4 zFactorCorrectedESurface(24)
    REAL*4 zFactorCorrectedNearSurface(24)
    REAL*4 paramNUBF(3,24)
    REAL*4 precipRate(88,24)
    REAL*4 precipWaterIntegrated(2,24)
    INTEGER*4 qualitySLV(24)
    REAL*4 precipRateNearSurface(24)
    REAL*4 precipRateESurface(24)
REAL*4 precipRateAve24(24)
CHARACTER phaseNearSurface(24)
REAL*4 epsilon(88,24)
END STRUCTURE

STRUCTURE /L2AKa_HS_EXPERIMENTAL/
  REAL*4 precipRateESurface2(24)
  CHARACTER precipRateESurface2Status(24)
  REAL*4 sigmaZeroProfile(5,24)
  INTEGER*2 binDEML2(24)
  REAL*4 seaIceConcentration(24)
END STRUCTURE

STRUCTURE /L2AKa_HS_DSD/
  CHARACTER phase(88,24)
  INTEGER*2 binNode(5,24)
END STRUCTURE

STRUCTURE /L2AKa_HS_SRT/
  REAL*4 PIAalt(6,24)
  REAL*4 RFactorAlt(6,24)
  REAL*4 PIAweight(6,24)
  REAL*4 pathAtten(24)
  REAL*4 reliabFactor(24)
  INTEGER*2 reliabFlag(24)
  INTEGER*2 refScanID(2,2,24)
  REAL*4 PIAhb(24)
  REAL*4 PIAhybrid(24)
  REAL*4 zeta(24)
  REAL*4 stddevEff(3,24)
  REAL*4 reliabFactorHY(24)
  REAL*4 stddevHY(24)
  INTEGER*2 reliabFlagHY(24)
END STRUCTURE

STRUCTURE /L2AKa_HS_CSF/
  INTEGER*4 flagBB(24)
  INTEGER*2 binBBPeak(24)
  INTEGER*2 binBBTop(24)
  INTEGER*2 binBBBottom(24)
  REAL*4 heightBB(24)
  REAL*4 widthBB(24)
  INTEGER*4 qualityBB(24)
INTEGER*4 typePrecip(24)
INTEGER*4 qualityTypePrecip(24)
INTEGER*4 flagShallowRain(24)
BYTE flagHeavyIcePrecip(24)
END STRUCTURE

STRUCTURE /L2AKa_HS_VER/
  INTEGER*2 binZeroDeg(24)
  REAL*4 attenuationNP(88,24)
  REAL*4 piaNP(4,24)
  REAL*4 sigmaZeroNPCorrected(24)
  REAL*4 heightZeroDeg(24)
END STRUCTURE

STRUCTURE /L2AKa_HS_PRE/
  REAL*4 elevation(24)
  INTEGER*4 landSurfaceType(24)
  REAL*4 localZenithAngle(24)
  INTEGER*4 flagPrecip(24)
  CHARACTER flagSigmaZeroSaturation(24)
  INTEGER*2 binRealSurface(24)
  INTEGER*2 binStormTop(24)
  REAL*4 heightStormTop(24)
  INTEGER*2 binClutterFreeBottom(24)
  REAL*4 sigmaZeroMeasured(24)
  REAL*4 zFactorMeasured(88,24)
  REAL*4 ellipsoidBinOffset(24)
  REAL*4 snRatioAtRealSurface(24)
  REAL*4 adjustFactor(24)
  BYTE snowIceCover(24)
END STRUCTURE

STRUCTURE /L2AKa_HS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
BYTE operationalMode
BYTE limitErrorFlag
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /L2AKa_HS/

RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(24)
REAL*4 Longitude(24)
RECORD /L2AKa_HS_SCANSTATUS/ scanStatus
RECORD /NAVIGATION/ navigation
RECORD /L2AKa_HS_PRE/ PRE
RECORD /L2AKa_HS_VER/ VER
RECORD /L2AKa_HS_CSF/ CSF
RECORD /L2AKa_HS_SRT/ SRT
RECORD /L2AKa_HS_DSD/ DSD
RECORD /L2AKa_HS_EXPERIMENTAL/ Experimental
RECORD /L2AKa_HS_SLV/ SLV
RECORD /L2AKa_HS_FLG/ FLG

END STRUCTURE

STRUCTURE /L2AKa_MS_FLG/

BYTE flagEcho(176,25)
INTEGER*4 qualityData(25)
BYTE qualityFlag(25)
BYTE flagSensor
INTEGER*2 flagScanPattern

END STRUCTURE

STRUCTURE /L2AKa_MS_SLV/

BYTE flagSLV(176,25)
REAL*4 paramDSD(2,176,25)
INTEGER*2 binEchoBottom(25)
REAL*4 piaFinal(25)
REAL*4 sigmaZeroCorrected(25)
REAL*4 zFactorCorrected(176,25)
REAL*4 zFactorCorrectedESurface(25)
REAL*4 zFactorCorrectedNearSurface(25)
REAL*4 paramNUBF(3,25)
REAL*4 precipRate(176,25)
REAL*4 precipWaterIntegrated(2,25)
INTEGER*4 qualitySLV(25)
REAL*4 precipRateNearSurface(25)
REAL*4 precipRateESurface(25)
REAL*4 precipRateAve24(25)
CHARACTER phaseNearSurface(25)
REAL*4 epsilon(176,25)
END STRUCTURE

STRUCTURE /L2AKa_MS_EXPERIMENTAL/
   REAL*4 precipRateESurface2(25)
   CHARACTER precipRateESurface2Status(25)
   REAL*4 sigmaZeroProfile(7,25)
   INTEGER*2 binDEML2(25)
   REAL*4 seaIceConcentration(25)
END STRUCTURE

STRUCTURE /L2AKa_MS_DSD/
   CHARACTER phase(176,25)
   INTEGER*2 binNode(5,25)
END STRUCTURE

STRUCTURE /L2AKa_MS_SRT/
   REAL*4 PIAalt(6,25)
   REAL*4 RFactorAlt(6,25)
   REAL*4 PIAweight(6,25)
   REAL*4 pathAtten(25)
   REAL*4 reliabFactor(25)
   INTEGER*2 reliabFlag(25)
   INTEGER*2 refScanID(2,2,25)
   REAL*4 PIAhb(25)
   REAL*4 PIAhybrid(25)
   REAL*4 zeta(25)
   REAL*4 stddevEff(3,25)
   REAL*4 reliabFactorHY(25)
   REAL*4 stddevHY(25)
   INTEGER*2 reliabFlagHY(25)
END STRUCTURE

STRUCTURE /L2AKa_MS_CSF/
   INTEGER*4 flagBB(25)
   INTEGER*2 binBBPeak(25)
   INTEGER*2 binBBTop(25)
   INTEGER*2 binBBBottom(25)
   REAL*4 heightBB(25)
   REAL*4 widthBB(25)
INTEGER*4 qualityBB(25)
INTEGER*4 typePrecip(25)
INTEGER*4 qualityTypePrecip(25)
INTEGER*4 flagShallowRain(25)
BYTE flagHeavyIcePrecip(25)

END STRUCTURE

STRUCTURE /L2AKa_MS_VER/
    INTEGER*2 binZeroDeg(25)
    REAL*4 attenuationNP(176,25)
    REAL*4 piaNP(4,25)
    REAL*4 sigmaZeroNPCorrected(25)
    REAL*4 heightZeroDeg(25)
END STRUCTURE

STRUCTURE /L2AKa_MS_PRE/
    REAL*4 elevation(25)
    INTEGER*4 landSurfaceType(25)
    REAL*4 localZenithAngle(25)
    INTEGER*4 flagPrecip(25)
    CHARACTER flagSigmaZeroSaturation(25)
    INTEGER*2 binRealSurface(25)
    INTEGER*2 binStormTop(25)
    REAL*4 heightStormTop(25)
    INTEGER*2 binClutterFreeBottom(25)
    REAL*4 sigmaZeroMeasured(25)
    REAL*4 zFactorMeasured(176,25)
    REAL*4 ellipsoidBinOffset(25)
    REAL*4 snRatioAtRealSurface(25)
    REAL*4 adjustFactor(25)
    BYTE snowIceCover(25)
END STRUCTURE

STRUCTURE /NAVIGATION/
    REAL*4 scPos(3)
    REAL*4 scVel(3)
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 dprAlt
    REAL*4 scAttRollGeoc
    REAL*4 scAttPitchGeoc
    REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L2AKa_MS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AKa_MS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(25)
  REAL*4 Longitude(25)
  RECORD /L2AKa_MS_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L2AKa_MS_PRE/ PRE
The DPR Level-2A product, 2ADPR, "DPR precipitation," is written as a 3 swath structure. The swaths are NS, normal scans, MS, matched scans, and HS, high sensitivity scans. The following sections describe the structure and contents of the format.

Dimension definitions:
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nscan</td>
<td>Number of scans in the granule.</td>
</tr>
<tr>
<td>nray</td>
<td>Number of angle bins in each NS scan.</td>
</tr>
<tr>
<td>nrayMS</td>
<td>Number of angle bins in each MS scan.</td>
</tr>
<tr>
<td>nrayHS</td>
<td>Number of angle bins in each HS scan.</td>
</tr>
<tr>
<td>nbin</td>
<td>Number of range bins in each NS and MS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.</td>
</tr>
<tr>
<td>nbinHS</td>
<td>Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.</td>
</tr>
<tr>
<td>nbinSZP</td>
<td>Number of range bins for sigmaZeroProfile.</td>
</tr>
<tr>
<td>nbinSZPHS</td>
<td>Number of range bins for sigmaZeroProfile in each HS scan.</td>
</tr>
<tr>
<td>nNP</td>
<td>Number of NP kinds.</td>
</tr>
<tr>
<td>nearFar</td>
<td>Near reference, Far reference.</td>
</tr>
<tr>
<td>foreBack</td>
<td>Forward, Backward.</td>
</tr>
<tr>
<td>method</td>
<td>Number of SRT methods.</td>
</tr>
<tr>
<td>nsdew</td>
<td>Number of standard deviation effective ways.</td>
</tr>
<tr>
<td>nNode</td>
<td>Number of binNode.</td>
</tr>
<tr>
<td>nDSD</td>
<td>Number of DSD parameters. Parameters are dBNw and Dm (mm).</td>
</tr>
<tr>
<td>LS</td>
<td>Liquid, solid.</td>
</tr>
<tr>
<td>nNUBF</td>
<td>Number of NUBF parameters.</td>
</tr>
<tr>
<td>two</td>
<td>Number of NUBF parameters.</td>
</tr>
<tr>
<td>three</td>
<td>Number of NUBF parameters.</td>
</tr>
<tr>
<td>thirty</td>
<td>Number of NUBF parameters.</td>
</tr>
<tr>
<td>thirteen</td>
<td>Number of NUBF parameters.</td>
</tr>
<tr>
<td>ten</td>
<td>Number of NUBF parameters.</td>
</tr>
<tr>
<td>six</td>
<td>Number of NUBF parameters.</td>
</tr>
<tr>
<td>four</td>
<td>Number of NUBF parameters.</td>
</tr>
<tr>
<td>eight</td>
<td>Number of NUBF parameters.</td>
</tr>
</tbody>
</table>

Figure 550 through Figure 588 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 550: Data Format Structure for 2ADPR, DPR precipitation

Figure 551: Data Format Structure for 2ADPR, NS
Figure 552: Data Format Structure for 2ADPR, MS
5.50 2ADPR - DPR precipitation

Figure 553: Data Format Structure for 2ADPR, HS

Figure 554: Data Format Structure for 2ADPR, NS, ScanTime
Figure 555: Data Format Structure for 2ADPR, NS, scanStatus

Figure 556: Data Format Structure for 2ADPR, NS, navigation
5.50 2ADPR - DPR precipitation

<table>
<thead>
<tr>
<th>Structure</th>
<th>Size</th>
<th>Array Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>elevation</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>landSurfaceType</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>localZenithAngle</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>flagPrecip</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>flagSigmaZeroSaturation</td>
<td>1 byte</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>binRealSurface</td>
<td>2 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>binStormTop</td>
<td>2 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>heightStormTop</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>binClutterFreeBottom</td>
<td>2 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>sigmaZeroMeasured</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>zFactorMeasured</td>
<td>4 bytes</td>
<td>nbin x nray x nscan</td>
</tr>
<tr>
<td>ellipsoidBinOffset</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>snRatioAtRealSurface</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>adjustFactor</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>snowIceCover</td>
<td>1 byte</td>
<td>nray x nscan</td>
</tr>
</tbody>
</table>

Figure 557: Data Format Structure for 2ADPR, NS, PRE

<table>
<thead>
<tr>
<th>Structure</th>
<th>Size</th>
<th>Array Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>binZeroDeg</td>
<td>2 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>attenuationNP</td>
<td>4 bytes</td>
<td>nbin x nray x nscan</td>
</tr>
<tr>
<td>piaNP</td>
<td>4 bytes</td>
<td>nNP x nray x nscan</td>
</tr>
<tr>
<td>sigmaZeroNPCorrected</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
<tr>
<td>heightZeroDeg</td>
<td>4 bytes</td>
<td>nray x nscan</td>
</tr>
</tbody>
</table>

Figure 558: Data Format Structure for 2ADPR, NS, VER
Figure 559: Data Format Structure for 2ADPR, NS, CSF

Figure 560: Data Format Structure for 2ADPR, NS, SRT
5.50 2ADPR - DPR precipitation

**Figure 561:** Data Format Structure for 2ADPR, NS, DSD

- DSD
  - phase: 1 byte, Array: nbin x nray x nscan
  - binNode: 2 bytes, Array: nNode x nray x nscan

**Figure 562:** Data Format Structure for 2ADPR, NS, Experimental

- Experimental
  - precipRateESurface2: 4 bytes, Array: nray x nscan
  - precipRateESurface2Status: 1 byte, Array: nray x nscan
  - sigmaZeroProfile: 4 bytes, Array: nbinSZP x nray x nscan
  - binDEML2: 2 bytes, Array: nray x nscan
  - seaIceConcentration: 4 bytes, Array: nray x nscan
Figure 563: Data Format Structure for 2ADPR, NS, SLV

Figure 564: Data Format Structure for 2ADPR, NS, FLG
5.50 2ADPR - DPR precipitation

ScanTime

- Year: 2 bytes, Array: nscan
- Month: 1 byte, Array: nscan
- DayOfMonth: 1 byte, Array: nscan
- Hour: 1 byte, Array: nscan
- Minute: 1 byte, Array: nscan
- Second: 1 byte, Array: nscan
- MilliSecond: 2 bytes, Array: nscan
- DayOfYear: 2 bytes, Array: nscan
- SecondOfDay: 8 bytes, Array: nscan

Figure 565: Data Format Structure for 2ADPR, MS, ScanTime

ScanStatus

- dataQuality: 1 byte, Array: nscan
- dataWarning: 1 byte, Array: nscan
- missing: 1 byte, Array: nscan
- modeStatus: 1 byte, Array: nscan
- geoError: 2 bytes, Array: nscan
- geoWarning: 2 bytes, Array: nscan
- SCorientation: 2 bytes, Array: nscan
- pointingStatus: 2 bytes, Array: nscan
- acsModeMidScan: 1 byte, Array: nscan
- targetSelectionMidScan: 1 byte, Array: nscan
- operationalMode: 1 byte, Array: nscan
- limitErrorFlag: 1 byte, Array: nscan
- FractionalGranuleNumber: 8 bytes, Array: nscan

Figure 566: Data Format Structure for 2ADPR, MS, scanStatus
Figure 567: Data Format Structure for 2ADPR, MS, navigation
5.50 2ADPR - DPR precipitation

**PRE**
- elevation 4 bytes Array: nrayMS x nsan
- landSurfaceType 4 bytes Array: nrayMS x nsan
- localZenithAngle 4 bytes Array: nrayMS x nsan
- flagPrecip 4 bytes Array: nrayMS x nsan
- flagSigmaZeroSaturation 1 byte Array: nrayMS x nsan
- binRealSurface 2 bytes Array: nrayMS x nsan
- binStormTop 2 bytes Array: nrayMS x nsan
- heightStormTop 4 bytes Array: nrayMS x nsan
- binClutterFreeBottom 2 bytes Array: nrayMS x nsan
- sigmaZeroMeasured 4 bytes Array: nrayMS x nsan
- zFactorMeasured 4 bytes Array: nbin x nrayMS x nsan
- ellipsoidBinOffset 4 bytes Array: nrayMS x nsan
- snRatioAtRealSurface 4 bytes Array: nrayMS x nsan
- adjustFactor 4 bytes Array: nrayMS x nsan
- snowIceCover 1 byte Array: nrayMS x nsan

Figure 568: Data Format Structure for 2ADPR, MS, PRE

**VER**
- binZeroDeg 2 bytes Array: nrayMS x nsan
- attenuationNP 4 bytes Array: nbin x nrayMS x nsan
- piaNP 4 bytes Array: nNP x nrayMS x nsan
- sigmaZeroNPCorrected 4 bytes Array: nrayMS x nsan
- heightZeroDeg 4 bytes Array: nrayMS x nsan

Figure 569: Data Format Structure for 2ADPR, MS, VER
Figure 570: Data Format Structure for 2ADPR, MS, CSF

Figure 571: Data Format Structure for 2ADPR, MS, SRT
Figure 572: Data Format Structure for 2ADPR, MS, DSD
Figure 573: Data Format Structure for 2ADPR, MS, Experimental

Figure 574: Data Format Structure for 2ADPR, MS, SLV

Figure 575: Data Format Structure for 2ADPR, MS, FLG
5.50 2ADPR - DPR precipitation

Figure 576: Data Format Structure for 2ADPR, TRG
continued from last figure

**TRG**

- triggerParameters 4 bytes  Array: \(8 \times n_{rayMS} \times n_{scan}\)

**Figure 577: Data Format Structure for 2ADPR, MS, TRG**

- **ScanTime**
  - Year 2 bytes  Array: \(n_{scan}\)
  - Month 1 byte  Array: \(n_{scan}\)
  - DayOfMonth 1 byte  Array: \(n_{scan}\)
  - Hour 1 byte  Array: \(n_{scan}\)
  - Minute 1 byte  Array: \(n_{scan}\)
  - Second 1 byte  Array: \(n_{scan}\)
  - MilliSecond 2 bytes  Array: \(n_{scan}\)
  - DayOfYear 2 bytes  Array: \(n_{scan}\)
  - SecondOfDay 8 bytes  Array: \(n_{scan}\)

**Figure 578: Data Format Structure for 2ADPR, HS, ScanTime**

- **scanStatus**
  - dataQuality 1 byte  Array: \(n_{scan}\)
  - dataWarning 1 byte  Array: \(n_{scan}\)
  - missing 1 byte  Array: \(n_{scan}\)
  - modeStatus 1 byte  Array: \(n_{scan}\)
  - geoError 2 bytes  Array: \(n_{scan}\)
  - geoWarning 2 bytes  Array: \(n_{scan}\)
  - SCorientation 2 bytes  Array: \(n_{scan}\)
  - pointingStatus 2 bytes  Array: \(n_{scan}\)
  - acsModeMidScan 1 byte  Array: \(n_{scan}\)
  - targetSelectionMidScan 1 byte  Array: \(n_{scan}\)
  - operationalMode 1 byte  Array: \(n_{scan}\)
  - limitErrorFlag 1 byte  Array: \(n_{scan}\)
  - FractionalGranuleNumber 8 bytes  Array: \(n_{scan}\)

**Figure 579: Data Format Structure for 2ADPR, HS, scanStatus**
Figure 580: Data Format Structure for 2ADPR, HS, navigation
Figure 581: Data Format Structure for 2ADPR, HS, PRE

Figure 582: Data Format Structure for 2ADPR, HS, VER
Figure 583: Data Format Structure for 2ADPR, HS, CSF
**STANDARD GPM PRODUCTS**

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
5.50 2ADPR - DPR precipitation

Experimental

- precipRateESurface2: 4 bytes, Array: nrayHS x nscan
- precipRateESurface2Status: 1 byte, Array: nrayHS x nscan
- sigmaZeroProfile: 4 bytes, Array: nbinSZPHS x nrayHS x nscan
- binDEML2: 2 bytes, Array: nrayHS x nscan
- seaIceConcentration: 4 bytes, Array: nrayHS x nscan

Figure 586: Data Format Structure for 2ADPR, HS, Experimental

SLV

- flagSLV: 1 byte, Array: nbinHS x nrayHS x nscan
- paramDSD: 4 bytes, Array: nDSD x nbinHS x nrayHS x nscan
- binEchoBottom: 2 bytes, Array: nrayHS x nscan
- piaFinal: 4 bytes, Array: nrayHS x nscan
- sigmaZeroCorrected: 4 bytes, Array: nrayHS x nscan
- zFactorCorrected: 4 bytes, Array: nrayHS x nscan
- zFactorCorrectedESurface: 4 bytes, Array: nrayHS x nscan
- zFactorCorrectedNearSurface: 4 bytes, Array: nrayHS x nscan
- paramNUBF: 4 bytes, Array: nNUBF x nrayHS x nscan
- precipRate: 4 bytes, Array: nbinHS x nrayHS x nscan
- precipWaterIntegrated: 4 bytes, Array: LS x nrayHS x nscan
- qualitySLV: 4 bytes, Array: nrayHS x nscan
- precipRateNearSurface: 4 bytes, Array: nrayHS x nscan
- precipRateESurface: 4 bytes, Array: nrayHS x nscan
- precipRateAve24: 4 bytes, Array: nrayHS x nscan
- phaseNearSurface: 1 byte, Array: nrayHS x nscan
- epsilon: 4 bytes, Array: nbinHS x nrayHS x nscan

Figure 587: Data Format Structure for 2ADPR, HS, SLV

FLG

- flagEcho: 1 byte, Array: nbinHS x nrayHS x nscan
- qualityData: 4 bytes, Array: nrayHS x nscan
- qualityFlag: 1 byte, Array: nrayHS x nscan
- flagSensor: 1 byte, Array: nscan
- flagScanPattern: 2 bytes, Array: nscan

Figure 588: Data Format Structure for 2ADPR, HS, FLG
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**NS** (Swath)

**NS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in NS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value
DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
  -9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
  -9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
  -9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
  -9999.9 Missing value

scanStatus (Group in NS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

  Bit Meaning if bit = 1
  0 missing
  5 geoError is not zero
  6 modeStatus is not zero

dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

  Bit Meaning if bit = 1
  0 Beam matching is abnormal
  1 VPRF table is abnormal
  2 Surface table is abnormal
  3 geoWarning is not zero
  4 Operational mode is not observation mode
  5 GPS status is abnormal
  6 Spare (always 0)
  7 Check sum of L1A is abnormal
missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

goError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. goError is used to set a bit in dataQuality. A zero integer value of goError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
</tbody>
</table>
1. Negative scan time, invalid input
2. Error getting spacecraft attitude at scan mid-time
3. Error getting spacecraft ephemeris at scan mid-time
4. Invalid input non-unit ray vector for any pixel
5. Ray misses Earth for any pixel with normal pointing
6. Nadir calculation error for subsatellite position
7. Pixel count with geolocation error over threshold
8. Error in getting spacecraft attitude for any pixel
9. Error in getting spacecraft ephemeris for any pixel
10. Spare (always 0)
11. Spare (always 0)
12. Spare (always 0)
13. Spare (always 0)
14. Spare (always 0)
15. Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
PointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
</tbody>
</table>
5.50 2ADPR - DPR precipitation

1. Flight Z axis nadir, +X in flight direction
2. S/C Z axis nadir, -X in flight direction
3. Flight Z axis nadir, -X in flight direction
4. +90 yaw for DPR antenna pattern calibration
5. -90 yaw for DPR antenna pattern calibration
-99 Missing

**operationalMode** (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

**limitErrorFlag** (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.
**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**navigation** (Group in NS)

**scPos** (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9  Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector (\(ms^{-1}\)) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9  Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9  Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital
Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value
timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980.
timeMidScan is used as the reference time for the scPos and scVel values. Values range
from 0 to 10000000000 s. Special values are defined as:
   -9999.9   Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0
to 100 s. Special values are defined as:
   -9999.9   Missing value

PRE (Group in NS)

elevation (4-byte float, array size: nray x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product.
Values are in m. Special values are defined as:
   -9999.9   Missing value

landSurfaceType (4-byte integer, array size: nray x nscan):

Land surface type.

0 - 99    Ocean
100 - 199  Land
200 - 299  Coast
300 - 399  Inland water
-9999      Missing value

localZenithAngle (4-byte float, array size: nray x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values
are in degree. Special values are defined as:
   -9999.9   Missing value

flagPrecip (4-byte integer, array size: nray x nscan):

Precipitation or no precipitation.

For L2 Ku and L2 Ka

0     No precipitation
1     Precipitation
-9999 Missing value
5.50  2ADPR - DPR precipitation

For L2 DPR

0   No precipitation by both Ku and Ka
1   Precipitation by Ka, no rain by Ku
10  Precipitation by Ku, no rain by Ka
11  Precipitation by both Ku and Ka
-9999 Missing value

flagSigmaZeroSaturation (1-byte char, array size: nray x nscan):

A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nray x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nray x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

heightStormTop (4-byte float, array size: nray x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value

binClutterFreeBottom (2-byte integer, array size: nray x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

sigmaZeroMeasured (4-byte float, array size: nray x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values
are in dB. Special values are defined as:
-9999.9 Missing value

**zFactorMeasured** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**ellipsoidBinOffset** (4-byte float, array size: nray x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

\[
\text{ellipsoidBinOffset} = \text{scRangeEllipsoid} - \{ \text{startBinRange} + (\text{binEllipsoid}-1) \times \text{rangeBinSize}\}
\]

**snRatioAtRealSurface** (4-byte float, array size: nray x nscan):
Signal/Noise ratio at real surface range bin.

\[
\text{snRatioAtRealSurface} = 10. \times \log_{10}(\text{echoPowertrueV}[\text{mW}] / \text{noisePowertrueV}[\text{mW}])
\]

-9999 Missing value

**adjustFactor** (4-byte float, array size: nray x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm') and sigmaZeroMeasured (dBs0m'). dBZm' and dBs0m' are used and stored as follows:

\[
d\text{BZm}' = d\text{BZm} - \text{adjustFactor} \\
d\text{Bs0m}' = d\text{Bs0m} - \text{adjustFactor}
\]

The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
 temporal adjustment for orbit number dependency.

**snowIceCover** (1-byte integer, array size: nray x nscan):
TBD. Special values are defined as:
-99 Missing value
VER (Group in NS)

binZeroDeg (2-byte integer, array size: nray x nscan):

Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

attenuationNP (4-byte float, array size: nbin x nray x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are defined as:
-9999.9 Missing value

piaNP (4-byte float, array size: nNP x nray x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are defined as:
-9999.9 Missing value

sigmaZeroNPCorrected (4-byte float, array size: nray x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

heightZeroDeg (4-byte float, array size: nray x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9 Missing value

CSF (Group in NS)
flagBB (4-byte integer, array size: nray x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand
and L2 Ku and L2 Ka on the other.

L2 DPR:
0    no Bright Band
1    Bright Band detected by Ku and DFRm
2    Bright Band detected by Ku only
3    Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value

L2 Ku and L2 Ka:
0    BB not detected
1    BB detected
-1111 No rain value
-9999 Missing value

binBBPeak (2-byte integer, array size: nray x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are
1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS
swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88
at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are
defined as:
-9999 Missing value

binBBTop (2-byte integer, array size: nray x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are
1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS
swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88
at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are
defined as:
-9999 Missing value

binBBBottom (2-byte integer, array size: nray x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are
1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS
swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88
at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are
defined as:
-9999 Missing value

heightBB (4-byte float, array size: nray x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m.
Special values are defined as:
-9999.9 Missing value
widthBB (4-byte float, array size: nray x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m.
Special values are defined as:
-9999.9 Missing value

qualityBB (4-byte integer, array size: nray x nscan):

Quality of the bright band.
When the bright band is detected,
a larger positive number indicates lower confidence in the detection.

The Ku detection is clear, but the Ka and DPR detection is somewhat doubtful.

The meaning of qualityBB has not been finalized.

3 Smeared bright band
2 Not so clear bright band
1 Clear bright band
0 BB not detected in the case of rain
-1111 No rain value
-9999 Missing value

typePrecip (4-byte integer, array size: nray x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, convective, and other, can be obtained as follows:

When typePrecip is greater than zero,
Major rain type = typePrecip/10000000
    = 1 stratiform
    = 2 convective
    = 3 other

-1111 No rain value
-9999 Missing value

Let abcdefgh be the 8 digit number,

    abcdefgh

then
a: Main rain type. (a=1,2,3),
b: 0,
c: 0,
d: V rain type,
e: H rain type,
f: BB,
g: Shallow rain,
h: Small size cell.

The following numbers appear as Ku and Ka (MS/HS) rain types:

---- stratiform
1001H100
10031000

---- convective
2001H1xy (x>0 or y>0)
2002Hbxy
200310xy (x>0 or y>0)
200320xy

---- other
300330xy

where H is the rain type by H-method, and b depends on BB,
x on shallow rain and y on small size cell:
H = 1: stratiform by H-method,
    2: convective by H-method,
    3: other by H-method.

b = 0: BB not detected,
    1: BB detected.

x = 0: No shallow rain,
    1: Shallow isolated,
    3: Shallow non-isolated.

y = 0: No small size cell,
    1: Single cell,
    2: Small size cell consisting of two adjacent pixels.

In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:

DFRm rain type = (typePrecip%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000 in FORTRAN
DFRm rain type
   = 1   stratiform
   = 2   convective
   = 4   transition
   = 8   DFRm method cannot be applicable at Part B (in this case
          the conventional method determines the major rain type)
   = 9   DFRm method cannot be applicable at Part A (in this case
          the conventional method determines the major rain type)

-1111  No rain value
-9999   Missing value

If dual frequency data is not available
but Ku-only or Ka-only is available,
rain type is expressed by the following 8 digit number:
   10xxxxxx --- stratiform,
   20xxxxxx --- convective,
   30xxxxxx --- other,
which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is
expressed by
   1qxxxxxx --- stratiform,
   2qxxxxxx --- convective,
   3qxxxxxx --- other,
where q>0.
Thus, by examining q, users can understand whether
data is processed by dual frequency algorithm or
single frequency algorithm.

For MS and HS, DFRm method is used.

DFRm decision classifies rain type into
   stratiform,
   convective,
and
   transition.

The DPR numbering rule can be summarized as follows:
Let opqrstuv be the 8 digit number, then
   o: Main rain type. (o=1,2,3),
   p: DFRm rain type. (p=0,1,2,4,8,9, with p=0 for single frequency data only),
   q: DFRm BB. (q=0,1),
r: V rain type (by conventional V-method).
   Basically r=0 for inner swath and r>0 for outer swath.
   However, r>0 when only single frequency data is available,

s: H rain type,
t: = 0 for inner swath,
   1 when BB is detected in the outer swath.
u: Shallow rain,
v: Small size cell.

DFRm type can be obtained by examining p

The meaning of p is as follows:
p = 0: single frequency data only (dual frequency data not available),
   1: stratiform by DFRm method,
   2: convective by DFRm method,
   4: transition by DFRm method,
   8: DFRm decision not available,
   9: DFRm decision not available.

Note that p>0 always in DPR processing, which is different
from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, p=0 always.

The following numbers appear as DPR rain types:

* For NS outer swath *

--- stratiform
1901H100
19031000
--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy
--- other
390330xy

* For NS inner swath and MS *

*******************

1590 5 STANDARD GPM PRODUCTS
--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
    or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type
    or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type
    or NS rain >0 but no MS rain; NS V and H determine rain type
--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
    or NS rain >0 but no MS rain; NS V and H determine rain type
    (x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
    or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
    (x>0 or y>0)
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
    or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
    or NS rain >0 but no MS rain; NS V and H determine rain type

**************
* For HS *
**************
--- stratiform
11B0H000
14B01000
19001000 --- H decision only
--- convective
21B0H0x0 (x>0)
where \( w \) depends on \( BB \) by conventional \( V \)-method, \( B \) on \( BB \) by DFRm method, \( H \) on \( H \)-method, \( x \) on shallow rain and \( y \) on small size cell:

- \( w = 0 \): \( BB \) not detected by conventional \( V \)-method,
  - 1: \( BB \) detected by conventional \( V \)-method.

- \( B = 0 \): \( BB \) not detected by DFRm method,
  - 1: \( BB \) detected by DFRm method.

- \( H = 1 \): stratiform by \( H \)-method,
  - 2: convective by \( H \)-method,
  - 3: other by \( H \)-method.

- \( x = 0 \): No shallow rain,
  - 1: Shallow isolated,
  - 3: Shallow non-isolated.

- \( y = 0 \): No small size cell,
  - 1: Single cell,
  - 2: Small size cell consisting of two adjacent pixels.

In the above, \( x>0 \) and \( y>0 \) are taken care of in the function \( R\_type\_classification\_dpr2() \).

=============================================

**qualityTypePrecip** (4-byte integer, array size: \( nray \times \) nscan):

Quality of the precipitation type.

1 Good
-1111 No rain value
-9999 Missing value

**flagShallowRain** (4-byte integer, array size: \( nray \times \) nscan):

22B0H0x0
240010x0 \( (x>0, \ 24B010x0 \ with \ B=0) \)
240020x0
241010x0 \( (x>0, \ 24B010x0 \ with \ B=1) \)
290010x0 \( (x>0) \ --- \ H \ decision \ only \)
290020x0 \( --- \ H \ decision \ only \)
--- other
340030x0
390030x0 \( --- \ H \ decision \ only \)
Type of shallow rain
0     No shallow rain
10    Shallow isolated (maybe)
11    Shallow isolated (certain)
20    Shallow non-isolated (maybe)
21    Shallow non-isolated (certain)
-1111 No rain value
-9999 Missing value

flagHeaveyIcePrecip (1-byte integer, array size: nray x nscan):
This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:
-99    Missing value

flagAnvil (1-byte integer, array size: nray x nscan):
flagAnvil is 1 when anvil is detected by the Ku-band radar,
0 when anvil is not detected, and
-99 when the data is missing.

Note that Ka-band decision is not made because of a lower sensitivity of Ka-band radar (therefore, there does not exist any Ka-band flagAnvil; only Ku-band flagAnvil is available in Ku-only and DPR NS).

SRT (Group in NS)

PIAalt (4-byte float, array size: method x nray x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where
PIAalt (j=1) = PIA_Ku from forward along-track spatial at kth angle bin
PIAalt (j=2) = PIA_Ku from backward along-track spatial at kth angle bin
PIAalt (j=3) = PIA_Ku from forward hybrid at kth angle bin
PIAalt (j=4) = PIA_Ku from backward hybrid at kth angle bin
PIAalt (j=5) = PIA_Ku from temporal reference at kth angle bin
PIAalt (j=6) = PIA_Ku from light-rain temporal reference at kth angle bin
Values are in dB. Special values are defined as:
-9999.9 Missing value

RFactorAlt (4-byte float, array size: method x nray x nscan):
The reliability factors associated with the individual PIA estimates corresponding to
PIAalt. Special values are defined as:
-9999.9 Missing value

**PIAweight** (4-byte float, array size: method x nray x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

\[
\text{PIAweight}_j = 1/\sigma_j^2 \times (1/\text{Sum}_j(1/\sigma_j^2))
\]

Special values are defined as:
-9999.9 Missing value

**pathAtten** (4-byte float, array size: nray x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
-9999.9 Missing value

**reliabFactor** (4-byte float, array size: nray x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
-9999.9 Missing value

**reliabFlag** (2-byte integer, array size: nray x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:
- \( = 1 \) if \( \text{Rel}_\text{eff} > 3 \); PIAeff estimate is considered reliable
- \( = 2 \) if \( 3 \geq \text{Rel}_\text{eff} > 1 \); PIAeff estimate is considered marginally reliable
- \( = 3 \) if \( \text{Rel}_\text{eff} \leq 1 \); PIAeff is unreliable
- \( = 4 \) if \( \text{SNR} \text{ at surface} < 2 \text{dB} \); provides a lower bound to the path-attenuation
- \( = 9 \) (no-rain case)

Special values are defined as:
-9999 Missing value

**refScanID** (2-byte integer, array size: nearFar x foreBack x nray x nscan):
The number of scan lines between the current scan and the beginning (or end) of the along-track reference data at each angle bin. The values are computed by the equation: Current Scan Number - Reference Scan Number. The values are positive for the Forward estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:

- \(1,1\) - Forward - Near reference
- \(2,1\) - Forward - Far reference
- \(1,2\) - Backward - Near reference
- \(2,2\) - Backward - Far reference

Special values are defined as:
-9999 Missing value
**PIAhb** (4-byte float, array size: nray x nscan):
The 2-way attenuation of HB.

Values are in dB. Special values are defined as:
-9999.9  Missing value

**PIAhybrid** (4-byte float, array size: nray x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as:
-9999.9  Missing value

**zeta** (4-byte float, array size: nray x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
-9999.9  Missing value

**stddevEff** (4-byte float, array size: nsdew x nray x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.

Special values are defined as:
-9999.9  Missing value

**reliabFactorHY** (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9  Missing value

**stddevHY** (4-byte float, array size: nray x nscan):
TBD.
Special values are defined as:
-9999.9 Missing value

**reliabFlagHY** (2-byte integer, array size: nray x nscan):
TBD.

Special values are defined as:
-9999 Missing value

**DSD** (Group in NS)

**phase** (1-byte char, array size: nbin x nray x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

- phase < 100 Temperature(C)=phase-100
- phase > 200 Temperature(C)=phase-200
- phase = 100 Top of the bright band
- phase = 200 Bottom of the bright band
- phase = 125 is used for the range bins between the top and peak of bright band
- phase = 175 is used for the range bins between the peak and bottom of bright band

Integer values of phase/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

**binNode** (2-byte integer, array size: nNode x nray x nscan):

The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
Convective: 750m below 0deg C level.

4 - Bin number of real surface equal to binRealSurface in PRE group.

For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid.
For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid.

-9999 - Missing

Experimental (Group in NS)

precipRateESurface2 (4-byte float, array size: nray x nscan):
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateESurface2Status (1-byte char, array size: nray x nscan):
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
255 Missing value

sigmaZeroProfile (4-byte float, array size: nbinSZP x nray x nscan):
Surface backscattering cross section profile around the current ifov. For information on this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:
-9999.9 Missing value

binDEML2 (2-byte integer, array size: nray x nscan):
Range bin number of the digital elevation model surface estimate. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
-9999 Missing value

seaIceConcentration (4-byte float, array size: nray x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value
SLV (Group in NS)

flagSLV (1-byte integer, array size: nbin x nray x nscan):
Special values are defined as:
-99  Missing value

paramDSD (4-byte float, array size: nDSD x nbin x nray x nscan):
Parameters of the drop size distribution. The first index is dBNw; the second index is Dm in mm. Special values are defined as:
-9999.9  Missing value

binEchoBottom (2-byte integer, array size: nray x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999  Missing value

piaFinal (4-byte float, array size: nray x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9  Missing value

sigmaZeroCorrected (4-byte float, array size: nray x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9  Missing value

zFactorCorrected (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9  Missing value

zFactorCorrectedESurface (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9  Missing value

zFactorCorrectedNearSurface (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9  Missing value

paramNUBF (4-byte float, array size: nNUBF x nray x nscan):
TBD. Special values are defined as:
-9999.9  Missing value

precipRate (4-byte float, array size: nbin x nray x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9  Missing value
precipWaterIntegrated (4-byte float, array size: LS x nray x nscan):
Precipitation water vertically integrated. Values are in g/m². Special values are defined as:
   -9999.9 Missing value

qualitySLV (4-byte integer, array size: nray x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are defined as:
   -9999 Missing value

precipRateNearSurface (4-byte float, array size: nray x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
   -9999.9 Missing value

precipRateESurface (4-byte float, array size: nray x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
   -9999.9 Missing value

precipRateAve24 (4-byte float, array size: nray x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
   -9999.9 Missing value

phaseNearSurface (1-byte char, array size: nray x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

phaseNearSurface < 100 Temperature(C)=phaseNearSurface-100
phaseNearSurface > 200 Temperature(C)=phaseNearSurface-200
phaseNearSurface = 100 Top of the bright band
phaseNearSurface = 200 Bottom of the bright band
phaseNearSurface = 125 is used for the range bins between the top and peak of bright band
phaseNearSurface = 175 is used for the range bins between the peak and bottom of bright band

Integer values of phaseNearSurface/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

epsilon (4-byte float, array size: nbin x nray x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution,
epsilon = 1 is no adjustment. Special values are defined as:
-9999.9  Missing value

**FLG (Group in NS)**

flagEcho (1-byte integer, array size: nbin x nray x nscan):

Flag of precipitation and main/side lobe clutter information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>

qualityData (4-byte integer, array size: nray x nscan):
Normal data gives "0". Non-zero values mean the kinds of errors. Special values are defined as:
-9999  Missing value

Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag has 2 bits of information.

The 2 bit flag for each module has values:
[higher bit  lower bit]
[0 0]   Good
[0 1]   Warning but usable
[1 0]   NG or error

The bits of qualityData are assigned as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 7</td>
<td>Copy of dataQuality in level 1B product</td>
</tr>
</tbody>
</table>
8 - 9 Flag by input module
10 - 11 Flag by preparation module
12 - 13 Flag by vertical module
14 - 15 Flag by classification module
16 - 17 Flag by SRT module
18 - 19 Flag by DSD module
20 - 21 Flag by solver module
22 - 23 Flag by output module
24 - 31 Spare

**qualityFlag** (1-byte integer, array size: nray x nscan):
Flag derived from qualityData with the following values: Special values are defined as:
-99 Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

**flagSensor** (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>

**flagScanPattern** (2-byte integer, array size: nscan):
Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**MS** (Swath)
**MS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in MS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: nrayMS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nrayMS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in MS)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
</tbody>
</table>
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).
Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
</tbody>
</table>
geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCOrientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SOrientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
</tbody>
</table>
pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks.
limitErrorFlag may be used in modeStatus.
Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group in MS)
scPos (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
   -9999.9  Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector ($ms^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
   -9999.9  Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
   -9999.9  Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9  Missing value

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
   -9999.9  Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
   -9999.9  Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9  Missing value
**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value
PRE (Group in MS)

elevation (4-byte float, array size: nrayMS x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:
-9999.9 Missing value

landSurfaceType (4-byte integer, array size: nrayMS x nscan):

Land surface type.

0 - 99 Ocean
100 - 199 Land
200 - 299 Coast
300 - 399 Inland water
-9999 Missing value

localZenithAngle (4-byte float, array size: nrayMS x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values are in degree. Special values are defined as:
-9999.9 Missing value

flagPrecip (4-byte integer, array size: nrayMS x nscan):

Precipitation or no precipitation.

For L2 Ku and L2 Ka

0 No precipitation
1 Precipitation
-9999 Missing value

For L2 DPR

0 No precipitation by both Ku and Ka
1 Precipitation by Ka, no rain by Ku
10 Precipitation by Ku, no rain by Ka
11 Precipitation by both Ku and Ka
-9999 Missing value
flagSigmaZeroSaturation (1-byte char, array size: nrayMS x nscan):

A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nrayMS x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nrayMS x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

heightStormTop (4-byte float, array size: nrayMS x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value

binClutterFreeBottom (2-byte integer, array size: nrayMS x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

sigmaZeroMeasured (4-byte float, array size: nrayMS x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:
-9999.9 Missing value

zFactorMeasured (4-byte float, array size: nbin x nrayMS x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:
-9999.9 Missing value

ellipsoidBinOffset (4-byte float, array size: nrayMS x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.
ellipsoidBinOffset =
    scRangeEllipsoid - \{ startBinRange + (binEllipsoid-1) \times rangeBinSize \}

scRangeEllipsoid : Distance between a sensor and the ellipsoid [m]
startBinRange : Distance between a sensor and a center
    of the highest observed range bin [m]
binEllipsoid : Range bin number of the Ellipsoid (1 - 260)
rangeBinSize : Range bin size [m]

-9999 Missing value

\textbf{snRatioAtRealSurface} (4-byte float, array size: nrayMS x nscan):
Signal/Noise ratio at real surface range bin.

\textbf{snRatioAtRealSurface} =
    10.*log10(echoPowertrueV[mW]/noisePowertrueV[mW])

-9999 Missing value

\textbf{adjustFactor} (4-byte float, array size: nrayMS x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm') and sigmaZeroMeasured (dBs0m').
\(\text{dBZm}' = \text{dBZm} - \text{adjustFactor} \)
\(\text{dBs0m}' = \text{dBs0m} - \text{adjustFactor} \)
The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
temporal adjustment for orbit number dependency.

\textbf{snowIceCover} (1-byte integer, array size: nrayMS x nscan):
TBD. Special values are defined as:
-99 Missing value

\textbf{VER} (Group in MS)

\textbf{binZeroDeg} (2-byte integer, array size: nrayMS x nscan):

Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

attenuationNP (4-byte float, array size: nbin x nrayMS x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud
ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are
defined as:
-9999.9 Missing value

piaNP (4-byte float, array size: nNP x nrayMS x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water,
cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are
defined as:
-9999.9 Missing value

sigmaZeroNPCorrected (4-byte float, array size: nrayMS x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation
particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

heightZeroDeg (4-byte float, array size: nrayMS x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9 Missing value

CSF (Group in MS)

flagBB (4-byte integer, array size: nrayMS x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand
and L2 Ku and L2 Ka on the other.

L2 DPR:
0 no Bright Band
1 Bright Band detected by Ku and DFRm
2 Bright Band detected by Ku only
3 Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value
L2 Ku and L2 Ka:
0 BB not detected
1 BB detected
-1111 No rain value
-9999 Missing value

**binBBPeak** (2-byte integer, array size: nrayMS x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binBBTop** (2-byte integer, array size: nrayMS x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binDFRmMLBottom** (2-byte integer, array size: nrayMS x nscan):
Range bin number for melting layer bottom detected by the DFRm method.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;0</td>
<td>Range bin number when ML bottom is detected</td>
</tr>
<tr>
<td>0</td>
<td>ML bottom not detected</td>
</tr>
<tr>
<td>-1111</td>
<td>Value for no rain in MS(HS) mode at Ka band</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**binDFRmMLTop** (2-byte integer, array size: nrayMS x nscan):
Range bin number for melting layer top detected by the DFRm method.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;0</td>
<td>Range bin number when ML top is detected</td>
</tr>
<tr>
<td>0</td>
<td>ML top not detected</td>
</tr>
<tr>
<td>-1111</td>
<td>Value for no rain in MS(HS) mode at Ka band</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**binBBBottom** (2-byte integer, array size: nrayMS x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values
are defined as:
-9999  Missing value

**heightBB** (4-byte float, array size: nrayMS x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9  Missing value

**widthBB** (4-byte float, array size: nrayMS x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9  Missing value

**qualityBB** (4-byte integer, array size: nrayMS x nscan):

Quality of the bright band.
When the bright band is detected,
a larger positive number indicates lower confidence in the detection.

The Ku detection is clear, but
the Ka and DPR detection is somewhat doubtful.

The meaning of qualityBB has not been finalized.

3  Smeared bright band
2  Not so clear bright band
1  Clear bright band
0  BB not detected in the case of rain
-1111  No rain value
-9999  Missing value

**typePrecip** (4-byte integer, array size: nrayMS x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, convective, and other, can be obtained as follows:

When typePrecip is greater than zero,
Major rain type = typePrecip/10000000
  = 1  stratiform
  = 2  convective
  = 3  other

-1111  No rain value
Let abcdefgh be the 8 digit number,

    abcdefgh

then

a: Main rain type. (a=1,2,3),
b: 0,
c: 0,
d: V rain type,
e: H rain type,
f: BB,
g: Shallow rain,
h: Small size cell.

The following numbers appear as Ku and Ka (MS/HS) rain types:

    ---- stratiform
    1001H100
    10031000
    ---- convective
    2001H1xy (x>0 or y>0)
    2002Hbxy
    200310xy (x>0 or y>0)
    200320xy
    ---- other
    300330xy

where H is the rain type by H-method, and b depends on BB,
x on shallow rain and y on small size cell:

H = 1: stratiform by H-method,
    2: convective by H-method,
    3: other by H-method.

b = 0: BB not detected,
    1: BB detected.

x = 0: No shallow rain,
    1: Shallow isolated,
    3: Shallow non-isolated.

y = 0: No small size cell,
    1: Single cell,
    2: Small size cell consisting of two adjacent pixels.
In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:

\[
\text{DFRm rain type} = \frac{\text{typePrecip} \mod 10000000}{1000000} \quad \text{in C}
\]
\[
\text{DFRm rain type} = \frac{\text{MOD}(\text{typePrecip},10000000)}{1000000} \quad \text{in FORTRAN}
\]

DFRm rain type
- 1 stratiform
- 2 convective
- 4 transition
- 8 DFRm method cannot be applicable at Part B (in this case the conventional method determines the major rain type)
- 9 DFRm method cannot be applicable at Part A (in this case the conventional method determines the major rain type)

-1111 No rain value
-9999 Missing value

If dual frequency data is not available but Ku-only or Ka-only is available, rain type is expressed by the following 8 digit number:
- 10xxxxxx --- stratiform,
- 20xxxxxx --- convective,
- 30xxxxxx --- other,
which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by
- 1qxxxxxx --- stratiform,
- 2qxxxxxx --- convective,
- 3qxxxxxx --- other,
where q>0.
Thus, by examining q, users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.

For MS and HS, DFRm method is used.

DFRm decision classifies rain type into
- stratiform,
- convective,
The DPR numbering rule can be summarized as follows:

Let opqrstuv be the 8 digit number, then

- **o**: Main rain type. (o=1,2,3),
- **p**: DFRm rain type. (p=0,1,2,4,8,9, with p=0 for single frequency data only),
- **q**: DFRm BB. (q=0,1),
- **r**: V rain type (by conventional V-method).
  
  Basically r=0 for inner swath and r>0 for outer swath.
  
  However, r>0 when only single frequency data is available,
- **s**: H rain type,
- **t**: = 0 for inner swath,
  
  1 when BB is detected in the outer swath.
- **u**: Shallow rain,
- **v**: Small size cell.

DFRm type can be obtained by examining p

The meaning of p is as follows:

- **p = 0**: single frequency data only (dual frequency data not available),
  
  1: stratiform by DFRm method,
  2: convective by DFRm method,
  4: transition by DFRm method,
  8: DFRm decision not available,
  9: DFRm decision not available.

Note that p>0 always in DPR processing, which is different from Ku-only or Ka-only result.

In Ku-only or Ka-only rain type numbering, p=0 always.

The following numbers appear as DPR rain types:

* For NS outer swath *

--- stratiform
1901H100
19031000

--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy
--- other
390330xy

*******************************
* For NS inner swath and MS *
*******************************
--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
(x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
(x>0 or y>0)
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type

**************
* For HS *
***************
--- stratiform
11B0H000
14B01000
19001000 --- H decision only
--- convective
21B0H0x0 (x>0)
22B0H0x0
240010x0 (x>0, 24B010x0 with B=0)
240020x0
241010x0 (x>0, 24B010x0 with B=1)
290010x0 (x>0) --- H decision only
290020x0 --- H decision only
--- other
340030x0
390030x0 --- H decision only

where w depends on BB by conventional V-method, B on BB by DFRm method, H on H-method, x on shallow rain and y on small size cell:
- \( w = 0 \): BB not detected by conventional V-method,
  1: BB detected by conventional V-method.
- \( B = 0 \): BB not detected by DFRm method,
  1: BB detected by DFRm method.
- \( H = 1 \): stratiform by H-method,
  2: convective by H-method,
  3: other by H-method.
- \( x = 0 \): No shallow rain,
  1: Shallow isolated,
  3: Shallow non-isolated.
- \( y = 0 \): No small size cell,
  1: Single cell,
  2: Small size cell consisting of two adjacent pixels.

In the above, \( x>0 \) and \( y>0 \) are taken care of in the function R\_type\_classification\_dpr2().

qualityTypePrecip (4-byte integer, array size: nrayMS x nscan):
Quality of the precipitation type.

1  Good
-1111  No rain value
-9999  Missing value

**flagShallowRain** (4-byte integer, array size: nrayMS x nscan):

Type of shallow rain
0  No shallow rain
10 Shallow isolated (maybe)
11 Shallow isolated (certain)
20 Shallow non-isolated (maybe)
21 Shallow non-isolated (certain)
-1111  No rain value
-9999  Missing value

**flagHeavyIcePrecip** (1-byte integer, array size: nrayMS x nscan):

This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:
-99  Missing value

**SRT** (Group in MS)

**PIAalt** (4-byte float, array size: method x nrayMS x nscan):

The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where

\[
\text{PIAalt (j=1)} = \text{PIA}_Ku \text{ from forward along-track spatial at kth angle bin}
\]
\[
\text{PIAalt (j=2)} = \text{PIA}_Ku \text{ from backward along-track spatial at kth angle bin}
\]
\[
\text{PIAalt (j=3)} = \text{PIA}_Ku \text{ from forward hybrid at kth angle bin}
\]
\[
\text{PIAalt (j=4)} = \text{PIA}_Ku \text{ from backward hybrid at kth angle bin}
\]
\[
\text{PIAalt (j=5)} = \text{PIA}_Ku \text{ from temporal reference at kth angle bin}
\]
\[
\text{PIAalt (j=6)} = \text{PIA}_Ku \text{ from light-rain temporal reference at kth angle bin}
\]

Values are in dB. Special values are defined as:
-9999.9  Missing value

**RFactorAlt** (4-byte float, array size: method x nrayMS x nscan):

The reliability factors associated with the individual PIA estimates corresponding to PIAalt. Special values are defined as:
-9999.9  Missing value
PIAweight (4-byte float, array size: method x nrayMS x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

\[ \text{PIAweight}_j = \frac{1}{\sigma_j^2} \times \left( \frac{1}{\sum_j (\frac{1}{\sigma_j^2})} \right) \]

Special values are defined as:
-9999.9 Missing value

pathAtten (4-byte float, array size: nrayMS x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
-9999.9 Missing value

reliabFactor (4-byte float, array size: nrayMS x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
-9999.9 Missing value

reliabFlag (2-byte integer, array size: nrayMS x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:
- 1 if Rel_eff > 3 ; PIAeff estimate is considered reliable
- 2 if 3 ≥ Rel_eff > 1 ; PIAeff estimate is considered marginally reliable
- 3 if Rel_eff ≤ 1 ; PIAeff is unreliable
- 4 if SNR_at surface < 2dB; provides a lower bound to the path-attenuation
- 9 (no-rain case)

Special values are defined as:
-9999 Missing value

refScanID (2-byte integer, array size: nearFar x foreBack x nrayMS x nscan):
The number of scan lines between the current scan and the beginning (or end) of the along-track reference data at each angle bin. The values are computed by the equation: Current Scan Number - Reference Scan Number. The values are positive for the Forward estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:

1,1 - Forward - Near reference
2,1 - Forward - Far reference
1,2 - Backward - Near reference
2,2 - Backward - Far reference

Special values are defined as:
-9999 Missing value

PIAhb (4-byte float, array size: nrayMS x nscan):
The 2-way attenuation of HB.
5.50  2ADPR - DPR precipitation

Values are in dB. Special values are defined as:
  -9999.9  Missing value

**PIAhybrid** (4-byte float, array size: nrayMS x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as:
  -9999.9  Missing value

**zeta** (4-byte float, array size: nrayMS x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
  -9999.9  Missing value

**stddevEff** (4-byte float, array size: nsdew x nrayMS x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.

Special values are defined as:
  -9999.9  Missing value

**reliabFactorHY** (4-byte float, array size: nrayMS x nscan):
TBD.

Special values are defined as:
  -9999.9  Missing value

**stddevHY** (4-byte float, array size: nrayMS x nscan):
TBD.

Special values are defined as:
  -9999.9  Missing value

**reliabFlagHY** (2-byte integer, array size: nrayMS x nscan):
TBD.
Special values are defined as:
-9999  Missing value

**DSD** (Group in MS)

**binNode** (2-byte integer, array size: nNode x nrayMS x nscan):

The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to
   binRealSurface in PRE group.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
-9999 - Missing

**Experimental** (Group in MS)

**precipRateESurface2** (4-byte float, array size: nrayMS x nscan):
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
-9999.9  Missing value

**precipRateESurface2Status** (1-byte char, array size: nrayMS x nscan):
Status of the estimated surface precipitation using alternate method. For information on
this experimental field contact the Joint DPR Team. Special values are defined as:

255  Missing value

**sigmaZeroProfile** (4-byte float, array size: nbinSZP x nrayMS x nscan):
Surface backscattering cross section profile around the current ifov. For information on this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:

-9999.9  Missing value

**binDEML2** (2-byte integer, array size: nrayMS x nscan):
Range bin number of the digital elevation model surface estimate. For information on this experimental field contact the Joint DPR Team. Special values are defined as:

-9999  Missing value

**seaIceConcentration** (4-byte float, array size: nrayMS x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:

-9999.9  Missing value

**flagSurfaceSnowfall** (1-byte char, array size: nrayMS x nscan):
Flag indicating snowfall on the surface, not aloft. 1 for snow, 0 for not snow. Special values are defined as:

255  Missing value

**surfaceSnowfallIndex** (4-byte float, array size: nrayMS x nscan):
Housekeeping product for test purposes. Special values are defined as:

-9999.9  Missing value

**SLV** (Group in MS)

**binEchoBottom** (2-byte integer, array size: nrayMS x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:

-9999  Missing value

**piaFinal** (4-byte float, array size: nrayMS x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:

-9999.9  Missing value

**sigmaZeroCorrected** (4-byte float, array size: nrayMS x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:

-9999.9  Missing value
**zFactorCorrected** (4-byte float, array size: nbin x nrayMS x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedESurface** (4-byte float, array size: nrayMS x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nrayMS x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**paramNUBF** (4-byte float, array size: nNUBF x nrayMS x nscan):
TBD. Special values are defined as:
-9999.9 Missing value

**precipWaterIntegrated** (4-byte float, array size: LS x nrayMS x nscan):
Precipitation water vertically integrated. Values are in g/m$^2$. Special values are defined as:
-9999.9 Missing value

**precipRateNearSurface** (4-byte float, array size: nrayMS x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface** (4-byte float, array size: nrayMS x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateAve24** (4-byte float, array size: nrayMS x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**phaseNearSurface** (1-byte char, array size: nrayMS x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

- $\text{phaseNearSurface} < 100 \quad \text{Temperature(C)} = \text{phaseNearSurface} - 100$
- $\text{phaseNearSurface} > 200 \quad \text{Temperature(C)} = \text{phaseNearSurface} - 200$
- $\text{phaseNearSurface} = 100 \quad \text{Top of the bright band}$
- $\text{phaseNearSurface} = 200 \quad \text{Bottom of the bright band}$
- $\text{phaseNearSurface} = 125 \quad \text{is used for the range bins between the top and peak of bright band}$
phaseNearSurface = 175 is used for the range bins between
the peak and bottom of bright band

Integer values of phaseNearSurface/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

epsilon (4-byte float, array size: nbin x nrayMS x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution,
epsilon = 1 is no adjustment. Special values are defined as:
-9999.9   Missing value

FLG (Group in MS)

flagEcho (1-byte integer, array size: nbin x nrayMS x nscan):

Flag of precipitation and main/side lobe clutter
information of each range bin.

Bit  Meaning
0  For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)
0  For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)
0  For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)
1  Precipitation judged by L2 DPR algorithm
2  Precipitation judged by L2 Ku algorithm
3  Precipitation judged by L2 Ka algorithm
4  Main lobe clutter judged by L2 Ku algorithm
5  Main lobe clutter judged by L2 Ka algorithm
6  Side lobe clutter judged by L2 Ku algorithm
7  Side lobe clutter judged by L2 Ka algorithm

qualityData (4-byte integer, array size: nrayMS x nscan):
Normal data gives ”0”. Non-zero values mean the kinds of errors. Special values are
defined as:
-9999   Missing value

Flag of quality data. Bit range from 8 to 23
contains flags by each module. Each module flag
has 2 bits of information.

The 2 bit flag for each module has values:
[higher bit  lower bit]
[0 0] Good
[0 1] Warning but usable
[1 0] NG or error

The bits of qualityData are assigned as follows:

Bit   Meaning
0 - 7   Copy of dataQuality in level 1B product
8 - 9   Flag by input module
10 - 11 Flag by preparation module
12 - 13 Flag by vertical module
14 - 15 Flag by classification module
16 - 17 Flag by SRT module
18 - 19 Flag by DSD module
20 - 21 Flag by solver module
22 - 23 Flag by output module
24 - 31 Spare

qualityFlag (1-byte integer, array size: nrayMS x nscan):
Flag derived from qualityData with the following values: Special values are defined as:
-99   Missing value

Value   Meaning
0       High quality. No issues.
1       Low quality (DPR modules had warnings but still made a retrieval)
2       Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)

flagSensor (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

Value   Meaning
1       Valid
-99     Invalid (judged by dataQuality)

flagScanPattern (2-byte integer, array size: nscan):
Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRG** (Group in MS)
This is an experimental part of the retrieval algorithm. Currently all fields within this group are set to zero.

**NUBFindex** (1-byte char, array size: nrayMS x nscan):

Trigger Primary Output: final index of NUBF presence.  
Integer between 0 and 100.  
This field currently set to all zero.

**MSindexKu** (1-byte char, array size: nrayMS x nscan):

Trigger Primary Output: final index of MS presence at Ku.  
Integer between 0 and 100.  
This field currently set to all zero.

**MSindexKa** (1-byte char, array size: nrayMS x nscan):

Trigger Primary Output: final index of MS presence at Ka.  
Integer between 0 and 100.  
This field currently set to all zero.

**precipFrac** (1-byte char, array size: three x nrayMS x nscan):

Trigger Primary Output: number of neighbors estimated to be "empty" in the 3 neighborhoods (4MS, 4MS+4HS, 8MS+4HS)  
This field currently set to all zero.

**RNUBFcond** (4-byte float, array size: nrayMS x nscan):

Trigger Primary Output: estimate of Sigma n  
(as defined in Iguchi et al. 2000)  
This field currently set to all zero.
MSsurfPeakIndexKu (1-byte char, array size: nrayMS x nscan):

Trigger Secondary Output: index of surface peak reliability for the purpose of MS assessment at Ku. This field currently set to all zero.

MSsurfPeakIndexKa (1-byte char, array size: nrayMS x nscan):

Trigger Secondary Output: index of surface peak reliability for the purpose of MS assessment at Ka. This field currently set to all zero.

MSthroughsurfIndexKu (1-byte char, array size: nrayMS x nscan):

Trigger Secondary Output: index of MS tail through surface at Ku. This field currently set to all zero.

MSthroughsurfIndexKa (1-byte char, array size: nrayMS x nscan):

Trigger Secondary Output: index of MS tail through surface at Ka. This field currently set to all zero.

MSkneeDFRindex (1-byte char, array size: nrayMS x nscan):

Trigger Secondary Output: index of DFR Knee presence. This field currently set to all zero.

MSthrZindex (1-byte char, array size: nrayMS x nscan):

Trigger Secondary Output: high Z in ice index. This field currently set to all zero.

NUBFratioPIAindex (1-byte char, array size: nrayMS x nscan):

Trigger Secondary Output: NUBF index based on the PIA departure. Accounts for PIA reliability flags. This field currently set to all zero.
NUBFnZmVarIndex (1-byte char, array size: three x nrayMS x nscan):

Trigger Secondary Output: NUBF index based on the variability of Z (flat weight) in 4/8/12 neighbors at Ka. This field currently set to all zero.

NUBFnZkVarIndex (1-byte char, array size: three x nrayMS x nscan):

Trigger Secondary Output: NUBF index based on the variability of Z (k-weighted) in 4/8/12 neighbors at Ka. This field currently set to all zero.

NUBFnZmVarScaling (2-byte integer, array size: nrayMS x nscan):

Trigger Secondary Output: scaling of the NUBFnZmVarIndex. This field currently set to all zero.

NUBFnZkVarScaling (2-byte integer, array size: nrayMS x nscan):

Trigger Secondary Output: scaling of the NUBFnZkVarIndex. This field currently set to all zero.

NUBFsurfSliceIndex (4-byte float, array size: thirty x nrayMS x nscan):

Placeholder for the Surface Range Slicing Approach by Meneghini and Liang. This field currently set to all zero.

NUBFprofZPC (4-byte float, array size: thirty x nrayMS x nscan):

Placeholder for the Z PC approach by Haddad. This field currently set to all zero.

MSbreakpoints (2-byte integer, array size: thirteen x nrayMS x nscan):

Trigger diagnostic. 3 range bins selected for the Knee check, and 5 for the through Surface check (for each Ku and Ka). This field currently set to all zero.
**MSslopes** (4-byte float, array size: ten x nrayMS x nscan):

Trigger diagnostic. 2 slopes for the Knee check, and 4 for the through Surface check. This field currently set to all zero.

**MSslopePoints** (4-byte float, array size: thirteen x nrayMS x nscan):

Trigger diagnostic. Zfit values at 13 critical breakpoints. This field currently set to all zero.

**MSslopeFits** (4-byte float, array size: six x nrayMS x nscan):

Trigger diagnostic. Rmse for the 5 slope fits. This field currently set to all zero.

**MSlowSNRrangeFilter** (1-byte char, array size: four x nrayMS x nscan):

Trigger diagnostic. Type and length of the 2 filters used to regularize profile below SNR. This field currently set to all zero.

**NUBFcorrPIA** (4-byte float, array size: two x nrayMS x nscan):

Trigger diagnostic. Final PIA after reconciliation, used for the NUBFratioPIAindex. This field currently set to all zero.

**triggerParameters** (4-byte float, array size: eight x nrayMS x nscan):

Trigger configuration. Set of tunable parameters (not output of the algorithm). Only for version control. This field currently set to all zero.

**HS** (Swath)
**HS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in HS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- -9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
- -99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
- -99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
- -99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
- -99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
- -99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
- -9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
- -9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
- -9999.9 Missing value

**Latitude** (4-byte float, array size: nrayHS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nrayHS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in HS)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
</tbody>
</table>
modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
</tbody>
</table>
geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
</tbody>
</table>
-8000 Non-nominal pointing
-9999 Missing

**pointingStatus** (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
<td></td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
<td></td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
<td></td>
</tr>
</tbody>
</table>

**acsModeMidScan** (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group in HS)
scPos (4-byte float, array size: XYZ x nscan):
The position vector(m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coor-
dinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan
period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector (ms⁻¹) of the spacecraft in ECEF Coordinates at the Scan mid-Time.
Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values
range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values
range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values
range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from
DPR science telemetry. This is empty in non-DPR products. Values range from 350000
to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The or-
der of the components in the file is roll, pitch, and yaw. However, the angles are computed
using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll
for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital
Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the
Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity
opposite the orbit normal direction, and the X-axis is approximately in the velocity
direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that
pitch and roll will have twice orbital frequency components due to the onboard control
system following the oblate geodetic Earth horizon. Note also that the yaw value will
show an orbital frequency component relative to the Earth fixed ground track due to the
Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees.
Special values are defined as:
-9999.9 Missing value
scAttPitchGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 1000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value
PRE (Group in HS)

elevation (4-byte float, array size: nrayHS x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:
   -9999.9   Missing value

landSurfaceType (4-byte integer, array size: nrayHS x nscan):

Land surface type.

0 - 99   Ocean
100 - 199 Land
200 - 299 Coast
300 - 399 Inland water
-9999   Missing value

localZenithAngle (4-byte float, array size: nrayHS x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values are in degree. Special values are defined as:
   -9999.9   Missing value

flagPrecip (4-byte integer, array size: nrayHS x nscan):

Precipitation or no precipitation.

For L2 Ku and L2 Ka

0   No precipitation
1   Precipitation
-9999   Missing value

For L2 DPR

0   No precipitation by both Ku and Ka
1   Precipitation by Ka, no rain by Ku
10  Precipitation by Ku, no rain by Ka
11  Precipitation by both Ku and Ka
-9999   Missing value
flagSigmaZeroSaturation (1-byte char, array size: nrayHS x nscan):

A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nrayHS x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nrayHS x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

heightStormTop (4-byte float, array size: nrayHS x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value

binClutterFreeBottom (2-byte integer, array size: nrayHS x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

sigmaZeroMeasured (4-byte float, array size: nrayHS x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:
-9999.9 Missing value

zFactorMeasured (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:
-9999.9 Missing value

ellipsoidBinOffset (4-byte float, array size: nrayHS x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.
ellipsoidBinOffset =
   scRangeEllipsoid - \{ startBinRange + (binEllipsoid-1) x rangeBinSize\}

scRangeEllipsoid  :  Distance between a sensor and the ellipsoid [m]
startBinRange     :  Distance between a sensor and a center of the highest observed range bin [m]
binEllipsoid      :  Range bin number of the Ellipsoid (1 - 260)
rangleBinSize     :  Range bin size [m]

-9999  Missing value

\textbf{snRatioAtRealSurface} (4-byte float, array size: nrayHS x nscan):
Signal/Noise ratio at real surface range bin.

\text{snRatioAtRealSurface} =
   10.*\log_{10}(\text{echoPowertrueV[mW]}/\text{noisePowertrueV[mW]})

-9999  Missing value

\textbf{adjustFactor} (4-byte float, array size: nrayHS x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm') and sigmaZeroMeasured (dBs0m').
dBZm' and dBs0m' are used and stored as follows:

dBZm' = dBZm - adjustFactor
dBs0m' = dBs0m - adjustFactor
The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
temporal adjustment for orbit number dependency.

\textbf{snowIceCover} (1-byte integer, array size: nrayHS x nscan):
TBD. Special values are defined as:
-99  Missing value

\textbf{VER} (Group in HS)

\textbf{binZeroDeg} (2-byte integer, array size: nrayHS x nscan):

Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window with 176 at the Ellipsoid.
For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

**attenuationNP** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are defined as:
-9999.9 Missing value

**piaNP** (4-byte float, array size: nNP x nrayHS x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are defined as:
-9999.9 Missing value

**sigmaZeroNPCorrected** (4-byte float, array size: nrayHS x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

**heightZeroDeg** (4-byte float, array size: nrayHS x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9 Missing value

**CSF** (Group in HS)

**flagBB** (4-byte integer, array size: nrayHS x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.

**L2 DPR:**
0 no Bright Band
1 Bright Band detected by Ku and DFRm
2 Bright Band detected by Ku only
3 Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value
L2 Ku and L2 Ka:
0      BB not detected
1      BB detected
-1111  No rain value
-9999  Missing value

**binBBPeak** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999  Missing value

**binBBTop** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999  Missing value

**binDFRmMLBottom** (2-byte integer, array size: nrayHS x nscan):
Range bin number for melting layer bottom detected by the DFRm method.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;0</td>
<td>Range bin number when ML bottom is detected</td>
</tr>
<tr>
<td>0</td>
<td>ML bottom not detected</td>
</tr>
<tr>
<td>-1111</td>
<td>Value for no rain in MS(HS) mode at Ka band</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**binDFRmMLTop** (2-byte integer, array size: nrayHS x nscan):
Range bin number for melting layer top detected by the DFRm method.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;0</td>
<td>Range bin number when ML top is detected</td>
</tr>
<tr>
<td>0</td>
<td>ML top not detected</td>
</tr>
<tr>
<td>-1111</td>
<td>Value for no rain in MS(HS) mode at Ka band</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**binBBBottom** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values
are defined as:
-9999  Missing value

**heightBB** (4-byte float, array size: nrayHS x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9  Missing value

**widthBB** (4-byte float, array size: nrayHS x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9  Missing value

**qualityBB** (4-byte integer, array size: nrayHS x nscan):

Quality of the bright band. When the bright band is detected, a larger positive number indicates lower confidence in the detection.

The Ku detection is clear, but the Ka and DPR detection is somewhat doubtful.

The meaning of qualityBB has not been finalized.

3  Smeared bright band
2  Not so clear bright band
1  Clear bright band
0  BB not detected in the case of rain
-1111  No rain value
-9999  Missing value

**typePrecip** (4-byte integer, array size: nrayHS x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, onvective, and other, can be obtained as follows:

When typePrecip is greater than zero,
Major rain type = typePrecip/10000000
   = 1  stratiform
   = 2  convective
   = 3  other

-1111  No rain value
-9999  Missing value

Let abcdefgh be the 8 digit number,

\[ abcdefgh \]

then

- a: Main rain type. (a=1,2,3),
- b: 0,
- c: 0,
- d: V rain type,
- e: H rain type,
- f: BB,
- g: Shallow rain,
- h: Small size cell.

The following numbers appear as Ku and Ka (MS/HS) rain types:

- ---- stratiform
  1001H100
  10031000
- ---- convective
  2001H1xy (x>0 or y>0)
  2002Hbxy
  200310xy (x>0 or y>0)
  200320xy
- ---- other
  300330xy

where H is the rain type by H-method, and b depends on BB,

- x on shallow rain and y on small size cell:
  - H = 1: stratiform by H-method,
  - 2: convective by H-method,
  - 3: other by H-method.

- b = 0: BB not detected,
  - 1: BB detected.

- x = 0: No shallow rain,
  - 1: Shallow isolated,
  - 3: Shallow non-isolated.

- y = 0: No small size cell,
  - 1: Single cell,
  - 2: Small size cell consisting of two adjacent pixels.
In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:

DFRm rain type = (typePrecip\%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000 in FORTRAN

DFRm rain type
  = 1  stratiform
  = 2  convective
  = 4  transition
  = 8  DFRm method cannot be applicable at Part B (in this case the conventional method determines the major rain type)
  = 9  DFRm method cannot be applicable at Part A (in this case the conventional method determines the major rain type)

-1111 No rain value
-9999 Missing value

If dual frequency data is not available but Ku-only or Ka-only is available, rain type is expressed by the following 8 digit number:
  10xxxxxx --- stratiform,
  20xxxxxx --- convective,
  30xxxxxx --- other,
which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by
  1qxxxxxx --- stratiform,
  2qxxxxxx --- convective,
  3qxxxxxx --- other,
where q>0.
Thus, by examining q, users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.

For MS and HS, DFRm method is used.

DFRm decision classifies rain type into
  stratiform,
  convective,
The DPR numbering rule can be summarized as follows:

Let opqrstuv be the 8 digit number, then

- **o**: Main rain type. \(o=1,2,3\),
- **p**: DFRm rain type. \(p=0,1,2,4,8,9\), with \(p=0\) for single frequency data only,
- **q**: DFRm BB. \(q=0,1\),
- **r**: V rain type (by conventional V-method).
  
  - Basically \(r=0\) for inner swath and \(r>0\) for outer swath.
  
  - However, \(r>0\) when only single frequency data is available,
- **s**: H rain type,
- **t**: = 0 for inner swath,
  
  - 1 when BB is detected in the outer swath.
- **u**: Shallow rain,
- **v**: Small size cell.

DFRm type can be obtained by examining **p**

The meaning of **p** is as follows:

- **p = 0**: single frequency data only (dual frequency data not available),
  
  - 1: stratiform by DFRm method,
  
  - 2: convective by DFRm method,
  
  - 4: transition by DFRm method,
  
  - 8: DFRm decision not available,
  
  - 9: DFRm decision not available.

Note that **p>0** always in DPR processing, which is different from Ku-only or Ka-only result.

In Ku-only or Ka-only rain type numbering, **p=0** always.

The following numbers appear as DPR rain types:

--- stratiform
1901H100
19031000

--- convective
2901H1xy \((x>0 \text{ or } y>0)\), see R\_type\_classification\_dpr2
2902Hwxy
290310xy \((x>0, y>0)\), see R\_type\_classification\_dpr2
290320xy
--- other
390330xy

*******************************
* For NS inner swath and MS *
*******************************

--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
          or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
          or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
          or NS rain >0 but no MS rain; NS V and H determine rain type.

--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
          or NS rain >0 but no MS rain; NS V and H determine rain type
          (x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
          or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
          or NS rain >0 but no MS rain; NS V and H determine rain type
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
          or NS rain >0 but no MS rain; NS V and H determine rain type

--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
          or NS rain >0 but no MS rain; NS V and H determine rain type

**************
* For HS *
where w depends on BB by conventional V-method, B on BB by DFRm method, H on H-method, x on shallow rain and y on small size cell:

- **w** = 0: BB not detected by conventional V-method,
  1: BB detected by conventional V-method.

- **B** = 0: BB not detected by DFRm method,
  1: BB detected by DFRm method.

- **H** = 1: stratiform by H-method,
  2: convective by H-method,
  3: other by H-method.

- **x** = 0: No shallow rain,
  1: Shallow isolated,
  3: Shallow non-isolated.

- **y** = 0: No small size cell,
  1: Single cell,
  2: Small size cell consisting of two adjacent pixels.

In the above, x>0 and y>0 are taken care of in the function `R\_type\_classification\_dpr2()`. 

------------------------
Quality of the precipitation type.

1 Good
-1111 No rain value
-9999 Missing value

flagShallowRain (4-byte integer, array size: nrayHS x nscan):

Type of shallow rain
0 No shallow rain
10 Shallow isolated (maybe)
11 Shallow isolated (certain)
20 Shallow non-isolated (maybe)
21 Shallow non-isolated (certain)
-1111 No rain value
-9999 Missing value

flagHeavyIcePrecip (1-byte integer, array size: nrayHS x nscan):
This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:
-99 Missing value

SRT (Group in HS)

PIAalt (4-byte float, array size: method x nrayHS x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where

\[
\text{PIAalt (j=1)} = \text{PIA}_Ku \text{ from forward along-track spatial at kth angle bin}
\]
\[
\text{PIAalt (j=2)} = \text{PIA}_Ku \text{ from backward along-track spatial at kth angle bin}
\]
\[
\text{PIAalt (j=3)} = \text{PIA}_Ku \text{ from forward hybrid at kth angle bin}
\]
\[
\text{PIAalt (j=4)} = \text{PIA}_Ku \text{ from backward hybrid at kth angle bin}
\]
\[
\text{PIAalt (j=5)} = \text{PIA}_Ku \text{ from temporal reference at kth angle bin}
\]
\[
\text{PIAalt (j=6)} = \text{PIA}_Ku \text{ from light-rain temporal reference at kth angle bin}
\]

Values are in dB. Special values are defined as:
-9999.9 Missing value

RFactorAlt (4-byte float, array size: method x nrayHS x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAalt. Special values are defined as:
-9999.9 Missing value
PIAweight (4-byte float, array size: method x nrayHS x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

\[
\text{PIAweight}_j = \frac{1}{\text{sigma}_j^2} \times \left( \frac{1}{\text{Sum}_j(1/\text{sigma}_j^2)} \right)
\]

Special values are defined as:
-9999.9 Missing value

pathAtten (4-byte float, array size: nrayHS x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
-9999.9 Missing value

reliabFactor (4-byte float, array size: nrayHS x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
-9999.9 Missing value

reliabFlag (2-byte integer, array size: nrayHS x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:
- 1 if Rel_eff > 3 ; PIAeff estimate is considered reliable
- 2 if 3 ≥ Rel_eff > 1 ; PIAeff estimate is considered marginally reliable
- 3 if Rel_eff ≤ 1 ; PIAeff is unreliable
- 4 if SNR_at surface < 2dB; provides a lower bound to the path-attenuation
- 9 (no-rain case)

Special values are defined as:
-9999 Missing value

refScanID (2-byte integer, array size: nearFar x foreBack x nrayHS x nscan):
The number of scan lines between the current scan and the beginning (or end) of the along-track reference data at each angle bin. The values are computed by the equation: Current Scan Number - Reference Scan Number. The values are positive for the Forward estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:

- 1,1 - Forward - Near reference
- 2,1 - Forward - Far reference
- 1,2 - Backward - Near reference
- 2,2 - Backward - Far reference

Special values are defined as:
-9999 Missing value

PIAhb (4-byte float, array size: nrayHS x nscan):
The 2-way attenuation of HB.
Values are in dB. Special values are defined as: 
-9999.9 Missing value

**PIAhybrid** (4-byte float, array size: nrayHS x nscan): 
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as: 
-9999.9 Missing value

**zeta** (4-byte float, array size: nrayHS x nscan): 
The term in the HB estimate of path attenuation.

Special values are defined as: 
-9999.9 Missing value

**stddevEff** (4-byte float, array size: nsdew x nrayHS x nscan): 
The effective standard deviation of PIA-SRT computed 3 ways.

Special values are defined as: 
-9999.9 Missing value

**reliabFactorHY** (4-byte float, array size: nrayHS x nscan): 
TBD.

Special values are defined as: 
-9999.9 Missing value

**stddevHY** (4-byte float, array size: nrayHS x nscan): 
TBD.

Special values are defined as: 
-9999.9 Missing value

**reliabFlagHY** (2-byte integer, array size: nrayHS x nscan): 
TBD.
Special values are defined as:
-9999 Missing value

**DSD (Group in HS)**

**phase** (1-byte char, array size: nbinHS x nrayHS x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

- `phase < 100` Temperature(C)=phase-100
- `phase > 200` Temperature(C)=phase-200
- `phase = 100` Top of the bright band
- `phase = 200` Bottom of the bright band
- `phase = 125` is used for the range bins between the top and peak of bright band
- `phase = 175` is used for the range bins between the peak and bottom of bright band

Integer values of `phase/100` =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

**binNode** (2-byte integer, array size: nNode x nrayHS x nscan):

The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to binRealSurface in PRE group.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
-9999 - Missing

**Experimental** (Group in HS)

**precipRateESurface2** (4-byte float, array size: nrayHS x nscan):
Estimates Surface Precipitation using alternate method. For information on this exper-
imental field contact the Joint DPR Team. Values are in mm/hr. Special values are
defined as:
-9999.9 Missing value

**precipRateESurface2Status** (1-byte char, array size: nrayHS x nscan):
Status of the estimated surface precipitation using alternate method. For information on
this experimental field contact the Joint DPR Team. Special values are defined as:
255 Missing value

**sigmaZeroProfile** (4-byte float, array size: nbinSZPHS x nrayHS x nscan):
Surface backscattering cross section profile around the current ifov. For information on
this experimental field contact the Joint DPR Team. Values are in dB. Special values are
defined as:
-9999.9 Missing value

**binDEML2** (2-byte integer, array size: nrayHS x nscan):
Range bin number of the digital elevation model surface estimate. For information on
this experimental field contact the Joint DPR Team. Special values are defined as:
-9999 Missing value

**seaIceConcentration** (4-byte float, array size: nrayHS x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact
the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value

**SLV** (Group in HS)

**flagSLV** (1-byte integer, array size: nbinHS x nrayHS x nscan):
Special values are defined as:
-99 Missing value
paramDSD (4-byte float, array size: nDSD x nbinHS x nrayHS x nscan): Parameters of the drop size distribution. The first index is dBNw; the second index is Dm in mm. Special values are defined as:
-9999.9 Missing value

binEchoBottom (2-byte integer, array size: nrayHS x nscan): For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

piaFinal (4-byte float, array size: nrayHS x nscan): The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

sigmaZeroCorrected (4-byte float, array size: nrayHS x nscan): Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9 Missing value

zFactorCorrected (4-byte float, array size: nbinHS x nrayHS x nscan): Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

zFactorCorrectedESurface (4-byte float, array size: nrayHS x nscan): Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

zFactorCorrectedNearSurface (4-byte float, array size: nrayHS x nscan): Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

paramNUBF (4-byte float, array size: nNUBF x nrayHS x nscan): TBD. Special values are defined as:
-9999.9 Missing value

precipRate (4-byte float, array size: nbinHS x nrayHS x nscan): Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipWaterIntegrated (4-byte float, array size: LS x nrayHS x nscan): Precipitation water vertically integrated. Values are in $g/m^2$. Special values are defined as:
-9999.9 Missing value
**qualitySLV** (4-byte integer, array size: nrayHS x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are defined as:
- 9999  Missing value

**precipRateNearSurface** (4-byte float, array size: nrayHS x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
- 9999.9  Missing value

**precipRateESurface** (4-byte float, array size: nrayHS x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
- 9999.9  Missing value

**precipRateAve24** (4-byte float, array size: nrayHS x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
- 9999.9  Missing value

**phaseNearSurface** (1-byte char, array size: nrayHS x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

- phaseNearSurface < 100  Temperature(C)=phaseNearSurface-100
- phaseNearSurface > 200  Temperature(C)=phaseNearSurface-200
- phaseNearSurface = 100  Top of the bright band
- phaseNearSurface = 200  Bottom of the bright band
- phaseNearSurface = 125  is used for the range bins between the top and peak of bright band
- phaseNearSurface = 175  is used for the range bins between the peak and bottom of bright band

Integer values of phaseNearSurface/100 =

- 0  - solid
- 1  - mixed phase
- 2  - liquid
- 255  - Missing

**epsilon** (4-byte float, array size: nbinHS x nrayHS x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution, 
epsilon = 1 is no adjustment. Special values are defined as:
- 9999.9  Missing value
**FLG (Group in HS)**

flagEcho (1-byte integer, array size: nbinHS x nrayHS x nscan):

Flag of precipitation and main/side lobe clutter information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>

**qualityData (4-byte integer, array size: nrayHS x nscan):**

Normal data gives "0". Non-zero values mean the kinds of errors. Special values are defined as:

-9999 Missing value

Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag has 2 bits of information.

The 2 bit flag for each module has values:

<table>
<thead>
<tr>
<th>[higher bit   lower bit]</th>
</tr>
</thead>
<tbody>
<tr>
<td>[0 0] Good</td>
</tr>
<tr>
<td>[0 1] Warning but usable</td>
</tr>
<tr>
<td>[1 0] NG or error</td>
</tr>
</tbody>
</table>

The bits of qualityData are assigned as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 7</td>
<td>Copy of dataQuality in level 1B product</td>
</tr>
<tr>
<td>8 - 9</td>
<td>Flag by input module</td>
</tr>
<tr>
<td>10 - 11</td>
<td>Flag by preparation module</td>
</tr>
<tr>
<td>12 - 13</td>
<td>Flag by vertical module</td>
</tr>
<tr>
<td>14 - 15</td>
<td>Flag by classification module</td>
</tr>
<tr>
<td>16 - 17</td>
<td>Flag by SRT module</td>
</tr>
</tbody>
</table>
18 - 19 Flag by DSD module
20 - 21 Flag by solver module
22 - 23 Flag by output module
24 - 31 Spare

qualityFlag (1-byte integer, array size: nrayHS x nscan):
Flag derived from qualityData with the following values: Special values are defined as:
-99    Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

flagSensor (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>

flagScanPattern (2-byte integer, array size: nscan):
Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

C Structure Header file:

```c
#ifndef _TK_2ADPR_H_
#define _TK_2ADPR_H_

#ifndef _L2ADPR_HS_FLG_
#define _L2ADPR_HS_FLG_

typedef struct {
    signed char flagEcho[24][88];
```
int qualityData[24];
signed char qualityFlag[24];
signed char flagSensor;
short flagScanPattern;
} L2ADPR_HS_FLG;
#endif

#ifndef _L2ADPR_HS_SLV_
#define _L2ADPR_HS_SLV_

typedef struct {
    signed char flagSLV[24][88];
    float paramDSD[24][88][2];
    short binEchoBottom[24];
    float piaFinal[24];
    float sigmaZeroCorrected[24];
    float zFactorCorrected[24][88];
    float zFactorCorrectedESurface[24];
    float zFactorCorrectedNearSurface[24];
    float paramNUBF[24][3];
    float precipRate[24][88];
    float precipWaterIntegrated[24][2];
    int qualitySLV[24];
    float precipRateNearSurface[24];
    float precipRateESurface[24];
    float precipRateAve24[24];
    unsigned char phaseNearSurface[24];
    float epsilon[24][88];
} L2ADPR_HS_SLV;
#endif

#ifndef _L2ADPR_HS_EXPERIMENTAL_
#define _L2ADPR_HS_EXPERIMENTAL_

typedef struct {
    float precipRateESurface2[24];
    unsigned char precipRateESurface2Status[24];
    float sigmaZeroProfile[24][5];
    short binDEML2[24];
    float seaIceConcentration[24];
} L2ADPR_HS_EXPERIMENTAL;
#ifndef _L2ADPR_HS_DSD_
#define _L2ADPR_HS_DSD_

typedef struct {
    unsigned char phase[24][88];
    short binNode[24][5];
} L2ADPR_HS_DSD;

#endif

#ifndef _L2ADPR_HS_SRT_
#define _L2ADPR_HS_SRT_

typedef struct {
    float PIAalt[24][6];
    float RFactorAlt[24][6];
    float PIAweight[24][6];
    float pathAtten[24];
    float reliabFactor[24];
    short reliabFlag[24];
    short refScanID[24][2][2];
    float PIAhb[24];
    float PIAhybrid[24];
    float zeta[24];
    float stddevEff[24][3];
    float reliabFactorHY[24];
    float stddevHY[24];
    short reliabFlagHY[24];
} L2ADPR_HS_SRT;

#endif

#ifndef _L2ADPR_HS_CSF_
#define _L2ADPR_HS_CSF_

typedef struct {
    int flagBB[24];
    short binBBPeak[24];
    short binBBTop[24];
    short binDFRmMLBottom[24];
} L2ADPR_HS_CSF;

#endif
typedef struct {
    short binDFRmMLTop[24];
    short binBBBottom[24];
    float heightBB[24];
    float widthBB[24];
    int qualityBB[24];
    int typePrecip[24];
    int qualityTypePrecip[24];
    int flagShallowRain[24];
    signed char flagHeavyIcePrecip[24];
} L2ADPR_HS_CSF;

#endif

#ifndef _L2ADPR_HS_VER_
#define _L2ADPR_HS_VER_

typedef struct {
    short binZeroDeg[24];
    float attenuationNP[24][88];
    float piaNP[24][4];
    float sigmaZeroNPCorrected[24];
    float heightZeroDeg[24];
} L2ADPR_HS_VER;

#endif

#ifndef _L2ADPR_HS_PRE_
#define _L2ADPR_HS_PRE_

typedef struct {
    float elevation[24];
    int landSurfaceType[24];
    float localZenithAngle[24];
    int flagPrecip[24];
    unsigned char flagSigmaZeroSaturation[24];
    short binRealSurface[24];
    short binStormTop[24];
    float heightStormTop[24];
    short binClutterFreeBottom[24];
    float sigmaZeroMeasured[24];
    float zFactorMeasured[24][88];
    float ellipsoidBinOffset[24];
    float snRatioAtRealSurface[24];
} L2ADPR_HS_PRE;

#endif
float adjustFactor[24];
    signed char snowIceCover[24];
} L2ADPR_HS_PRE;

#endif

#ifndef _L2ADPR_HS_SCANSTATUS_
#define _L2ADPR_HS_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2ADPR_HS_SCANSTATUS;

#endif

#ifndef _L2ADPR_HS_
#define _L2ADPR_HS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[24];
    float Longitude[24];
    L2ADPR_HS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2ADPR_HS_PRE PRE;
    L2ADPR_HS_VER VER;
    L2ADPR_HS_CSF CSF;
    L2ADPR_HS_SRT SRT;
    L2ADPR_HS_DSD DSD;
    L2ADPR_HS_EXPERIMENTAL Experimental;
    L2ADPR_HS_SLV SLV;
} L2ADPR_HS;
typedef struct {
    unsigned char NUBFindex[25];
    unsigned char MSindexKu[25];
    unsigned char MSindexKa[25];
    unsigned char precipFrac[25][3];
    float RNUBFcond[25];
    unsigned char MSsurfPeakIndexKu[25];
    unsigned char MSsurfPeakIndexKa[25];
    unsigned char MStroughsurfIndexKu[25];
    unsigned char MStroughsurfIndexKa[25];
    unsigned char MSkneeDFRindex[25];
    unsigned char MSthrZindex[25];
    unsigned char NUBFratioPIAindex[25];
    unsigned char NUBFnZmVarIndex[25][3];
    unsigned char NUBFnZkVarIndex[25][3];
    short NUBFnZmVarScaling[25];
    short NUBFnZkVarScaling[25];
    float NUBFsurfSliceIndex[25][30];
    float NUBFprofZPC[25][30];
    short MSbreakpoints[25][13];
    float MSslopes[25][10];
    float MSslopePoints[25][13];
    float MSslopeFits[25][6];
    unsigned char MSlowSNRrangeFilter[25][4];
    float NUBFcorrPIA[25][2];
    float triggerParameters[25][8];
} L2ADPR_MS_TRG;

#endif

#endif

typedef struct {
    signed char flagEcho[25][176];
int qualityData[25];
signed char qualityFlag[25];
signed char flagSensor;
short flagScanPattern;
} L2ADPR_MS_FLG;

#endif

#ifndef _L2ADPR_MS_SLV_
define _L2ADPR_MS_SLV_

typedef struct {
    short binEchoBottom[25];
    float piaFinal[25];
    float sigmaZeroCorrected[25];
    float zFactorCorrected[25][176];
    float zFactorCorrectedESurface[25];
    float zFactorCorrectedNearSurface[25];
    float paramNUBF[25][3];
    float precipWaterIntegrated[25][2];
    float precipRateNearSurface[25];
    float precipRateESurface[25];
    float precipRateAve24[25];
    unsigned char phaseNearSurface[25];
    float epsilon[25][176];
} L2ADPR_MS_SLV;

#endif

#ifndef _L2ADPR_MS_EXPERIMENTAL_
define _L2ADPR_MS_EXPERIMENTAL_

typedef struct {
    float precipRateESurface2[25];
    unsigned char precipRateESurface2Status[25];
    float sigmaZeroProfile[25][7];
    short binDEML2[25];
    float seaIceConcentration[25];
    unsigned char flagSurfaceSnowfall[25];
    float surfaceSnowfallIndex[25];
} L2ADPR_MS_EXPERIMENTAL;

#endif
#ifndef _L2ADPR_MS_DSD_
define _L2ADPR_MS_DSD_

typedef struct {
    short binNode[25][5];
} L2ADPR_MS_DSD;
#endif

#ifndef _L2ADPR_MS_SRT_
define _L2ADPR_MS_SRT_

typedef struct {
    float PIAalt[25][6];
    float RFactorAlt[25][6];
    float PIAweight[25][6];
    float pathAtten[25];
    float reliabFactor[25];
    short reliabFlag[25];
    short refScanID[25][2][2];
    float PIAhb[25];
    float PIAhybrid[25];
    float zeta[25];
    float stddevEff[25][3];
    float reliabFactorHY[25];
    float stddevHY[25];
    short reliabFlagHY[25];
} L2ADPR_MS_SRT;
#endif

#ifndef _L2ADPR_MS_CSF_
define _L2ADPR_MS_CSF_

typedef struct {
    int flagBB[25];
    short binBBPeak[25];
    short binBBTop[25];
    short binDFRmMLBottom[25];
    short binDFRmMLTop[25];
    short binBBBottom[25];
    float heightBB[25];
} L2ADPR_MS_CSF;
#endif
float widthBB[25];
int qualityBB[25];
int typePrecip[25];
int qualityTypePrecip[25];
int flagShallowRain[25];
signed char flagHeavyIcePrecip[25];
} L2ADPR_MS_CSF;

#endif

#ifndef _L2ADPR_MS_VER_
#define _L2ADPR_MS_VER_

typedef struct {
    short binZeroDeg[25];
    float attenuationNP[25][176];
    float piaNP[25][4];
    float sigmaZeroNPCorrected[25];
    float heightZeroDeg[25];
} L2ADPR_MS_VER;

#endif

#ifndef _L2ADPR_MS_PRE_
#define _L2ADPR_MS_PRE_

typedef struct {
    float elevation[25];
    int landSurfaceType[25];
    float localZenithAngle[25];
    int flagPrecip[25];
    unsigned char flagSigmaZeroSaturation[25];
    short binRealSurface[25];
    short binStormTop[25];
    float heightStormTop[25];
    short binClutterFreeBottom[25];
    float sigmaZeroMeasured[25];
    float zFactorMeasured[25][176];
    float ellipsoidBinOffset[25];
    float snRatioAtRealSurface[25];
    float adjustFactor[25];
    signed char snowIceCover[25];
} L2ADPR_MS_PRE;
typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2ADPR_MS_SCANSTATUS;

} L2ADPR_MS_SCANSTATUS;

typedef struct {
    SCANTIME ScanTime;
    float Latitude[25];
    float Longitude[25];
    L2ADPR_MS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2ADPR_MS_PRE PRE;
    L2ADPR_MS_VER VER;
    L2ADPR_MS_CSF CSF;
    L2ADPR_MS_SRT SRT;
    L2ADPR_MS_DSD DSD;
    L2ADPR_MS_EXPERIMENTAL Experimental;
    L2ADPR_MS_SLV SLV;
    L2ADPR_MS_FLG FLG;
    L2ADPR_MS_TRG TRG;
} L2ADPR_MS;
#ifndef _L2ADPR_NS_FLG_
#define _L2ADPR_NS_FLG_

typedef struct {
    signed char flagEcho[49][176];
    int qualityData[49];
    signed char qualityFlag[49];
    signed char flagSensor;
    short flagScanPattern;
} L2ADPR_NS_FLG;

#endif

#ifndef _L2ADPR_NS_SLV_
#define _L2ADPR_NS_SLV_

typedef struct {
    signed char flagSLV[49][176];
    float paramDSD[49][176][2];
    short binEchoBottom[49];
    float piaFinal[49];
    float sigmaZeroCorrected[49];
    float zFactorCorrected[49][176];
    float zFactorCorrectedESurface[49];
    float zFactorCorrectedNearSurface[49];
    float paramNUBF[49][3];
    float precipRate[49][176];
    float precipWaterIntegrated[49][2];
    int qualitySLV[49];
    float precipRateNearSurface[49];
    float precipRateESurface[49];
    float precipRateAve24[49];
    unsigned char phaseNearSurface[49];
    float epsilon[49][176];
} L2ADPR_NS_SLV;

#endif

#ifndef _L2ADPR_NS_EXPERIMENTAL_
#define _L2ADPR_NS_EXPERIMENTAL_
typedef struct {
    float precipRateESurface2[49];
    unsigned char precipRateESurface2Status[49];
    float sigmaZeroProfile[49][7];
    short binDEML2[49];
    float seaIceConcentration[49];
} L2ADPR_NS_EXPERIMENTAL;

#endif

#ifndef _L2ADPR_NS_DSD_
#define _L2ADPR_NS_DSD_

typedef struct {
    unsigned char phase[49][176];
    short binNode[49][5];
} L2ADPR_NS_DSD;
#endif

#ifndef _L2ADPR_NS_SRT_
#define _L2ADPR_NS_SRT_

typedef struct {
    float PIAalt[49][6];
    float RFactorAlt[49][6];
    float PIAweight[49][6];
    float pathAtten[49];
    float reliabFactor[49];
    short reliabFlag[49];
    short refScanID[49][2][2];
    float PIAhb[49];
    float PIAhybrid[49];
    float zeta[49];
    float stddevEff[49][3];
    float reliabFactorHY[49];
    float stddevHY[49];
    short reliabFlagHY[49];
} L2ADPR_NS_SRT;
#endif
#ifndef _L2ADPR_NS_CSF_
#define _L2ADPR_NS_CSF_

typedef struct {
    int flagBB[49];
    short binBBPeak[49];
    short binBBTop[49];
    short binBBBottom[49];
    float heightBB[49];
    float widthBB[49];
    int qualityBB[49];
    int typePrecip[49];
    int qualityTypePrecip[49];
    int flagShallowRain[49];
    signed char flagHeavyIcePrecip[49];
    signed char flagAnvil[49];
} L2ADPR_NS_CSF;

#endif

#ifndef _L2ADPR_NS_VER_
#define _L2ADPR_NS_VER_

typedef struct {
    short binZeroDeg[49];
    float attenuationNP[49][176];
    float piaNP[49][4];
    float sigmaZeroNPCorrected[49];
    float heightZeroDeg[49];
} L2ADPR_NS_VER;

#endif

#ifndef _L2ADPR_NS_PRE_
#define _L2ADPR_NS_PRE_

typedef struct {
    float elevation[49];
    int landSurfaceType[49];
    float localZenithAngle[49];
    int flagPrecip[49];
    unsigned char flagSigmaZeroSaturation[49];
    short binRealSurface[49];
} L2ADPR_NS_PRE;

#endif
short binStormTop[49];
float heightStormTop[49];
short binClutterFreeBottom[49];
float sigmaZeroMeasured[49];
float zFactorMeasured[49][176];
float ellipsoidBinOffset[49];
float snRatioAtRealSurface[49];
float adjustFactor[49];
signed char snowIceCover[49];
}

L2ADPR_NS_PRE;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
  float scPos[3];
  float scVel[3];
  float scLat;
  float scLon;
  float scAlt;
  float dprAlt;
  float scAttRollGeoc;
  float scAttPitchGeoc;
  float scAttYawGeoc;
  float scAttRollGeod;
  float scAttPitchGeod;
  float scAttYawGeod;
  float greenHourAng;
  double timeMidScan;
  double timeMidScanOffset;
} NAVIGATION;

#endif

#ifndef _L2ADPR_NS_SCANSTATUS_
#define _L2ADPR_NS_SCANSTATUS_

typedef struct {
  signed char dataQuality;
  signed char dataWarning;
  signed char missing;
} L2ADPR_NS_SCANSTATUS;

#endif
signed char modeStatus;
short geoError;
short geoWarning;
short SCorientation;
short pointingStatus;
signed char acsModeMidScan;
signed char targetSelectionMidScan;
signed char operationalMode;
signed char limitErrorFlag;
double FractionalGranuleNumber;
}
}

#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L2ADPR_NS_
#define _L2ADPR_NS_

typedef struct {
    SCANTIME ScanTime;
float Latitude[49];
float Longitude[49];
L2ADPR_NS_SCANSTATUS scanStatus;
NAVIGATION navigation;
L2ADPR_NS_PRE PRE;
L2ADPR_NS_VER VER;
L2ADPR_NS_CSF CSF;
} L2ADPR_NS_SCANSTATUS;

#endif
L2ADPR_NS_SRT SRT;
L2ADPR_NS_DSD DSD;
L2ADPR_NS_EXPERIMENTAL Experimental;
L2ADPR_NS_SLV SLV;
L2ADPR_NS_FLG FLG;
} L2ADPR_NS;

#endif

#ifndef _L2ADPR_SWATHS_
#define _L2ADPR_SWATHS_

typedef struct {
    L2ADPR_NS NS;
    L2ADPR_MS MS;
    L2ADPR_HS HS;
} L2ADPR_SWATHS;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L2ADPR_HS_FLG/
    BYTE flagEcho(88,24)
    INTEGER*4 qualityData(24)
    BYTE qualityFlag(24)
    BYTE flagSensor
    INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2ADPR_HS_SLV/
    BYTE flagSLV(88,24)
    REAL*4 paramDSD(2,88,24)
    INTEGER*2 binEchoBottom(24)
    REAL*4 piaFinal(24)
    REAL*4 sigmaZeroCorrected(24)
    REAL*4 zFactorCorrected(88,24)
    REAL*4 zFactorCorrectedESurface(24)
    REAL*4 zFactorCorrectedNearSurface(24)
    REAL*4 paramNUBF(3,24)
    REAL*4 precipRate(88,24)
REAL*4 precipWaterIntegrated(2,24)
INTEGER*4 qualitySLV(24)
REAL*4 precipRateNearSurface(24)
REAL*4 precipRateESurface(24)
REAL*4 precipRateAve24(24)
CHARACTER phaseNearSurface(24)
REAL*4 epsilon(88,24)
END STRUCTURE

STRUCTURE /L2ADPR_HS_EXPERIMENTAL/
  REAL*4 precipRateESurface2(24)
  CHARACTER precipRateESurface2Status(24)
  REAL*4 sigmaZeroProfile(5,24)
  INTEGER*2 binDEML2(24)
  REAL*4 seaIceConcentration(24)
END STRUCTURE

STRUCTURE /L2ADPR_HS_DSD/
  CHARACTER phase(88,24)
  INTEGER*2 binNode(5,24)
END STRUCTURE

STRUCTURE /L2ADPR_HS_SRT/
  REAL*4 PIAalt(6,24)
  REAL*4 RFactorAlt(6,24)
  REAL*4 PIAweight(6,24)
  REAL*4 pathAtten(24)
  REAL*4 reliabFactor(24)
  INTEGER*2 reliabFlag(24)
  INTEGER*2 refScanID(2,2,24)
  REAL*4 PIAhb(24)
  REAL*4 PIAhybrid(24)
  REAL*4 zeta(24)
  REAL*4 stddevEff(3,24)
  REAL*4 reliabFactorHY(24)
  REAL*4 stddevHY(24)
  INTEGER*2 reliabFlagHY(24)
END STRUCTURE

STRUCTURE /L2ADPR_HS_CSF/
  INTEGER*4 flagBB(24)
  INTEGER*2 binBBPeak(24)
  INTEGER*2 binBBTop(24)
END STRUCTURE
INTEGER*2 binDFRmMLBottom(24)
INTEGER*2 binDFRmMLTop(24)
INTEGER*2 binBBBottom(24)
REAL*4 heightBB(24)
REAL*4 widthBB(24)
INTEGER*4 qualityBB(24)
INTEGER*4 typePrecip(24)
INTEGER*4 qualityTypePrecip(24)
INTEGER*4 flagShallowRain(24)
BYTE flagHeavyIcePrecip(24)
END STRUCTURE

STRUCTURE /L2ADPR_HS_VER/
  INTEGER*2 binZeroDeg(24)
  REAL*4 attenuationNP(88,24)
  REAL*4 piaNP(4,24)
  REAL*4 sigmaZeroNPCorrected(24)
  REAL*4 heightZeroDeg(24)
END STRUCTURE

STRUCTURE /L2ADPR_HS_PRE/
  REAL*4 elevation(24)
  INTEGER*4 landSurfaceType(24)
  REAL*4 localZenithAngle(24)
  INTEGER*4 flagPrecip(24)
  CHARACTER flagSigmaZeroSaturation(24)
  INTEGER*2 binRealSurface(24)
  INTEGER*2 binStormTop(24)
  REAL*4 heightStormTop(24)
  INTEGER*2 binClutterFreeBottom(24)
  REAL*4 sigmaZeroMeasured(24)
  REAL*4 zFactorMeasured(88,24)
  REAL*4 ellipsoidBinOffset(24)
  REAL*4 snRatioAtRealSurface(24)
  REAL*4 adjustFactor(24)
  BYTE snowIceCover(24)
END STRUCTURE

STRUCTURE /L2ADPR_HS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
BYTE limitErrorFlag
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /L2ADPR_HS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(24)
  REAL*4 Longitude(24)
  RECORD /L2ADPR_HS_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L2ADPR_HS_PRE/ PRE
  RECORD /L2ADPR_HS_VER/ VER
  RECORD /L2ADPR_HS_CSF/ CSF
  RECORD /L2ADPR_HS_SRT/ SRT
  RECORD /L2ADPR_HS_DSD/ DSD
  RECORD /L2ADPR_HS_EXPERIMENTAL/ Experimental
  RECORD /L2ADPR_HS_SLV/ SLV
  RECORD /L2ADPR_HS_FLG/ FLG
END STRUCTURE

STRUCTURE /L2ADPR_MS_TRG/
  CHARACTER NUBFindex(25)
  CHARACTER MSindexKu(25)
  CHARACTER MSindexKa(25)
  CHARACTER precipFrac(3,25)
  REAL*4 RNUBFcond(25)
  CHARACTER MSSurfPeakIndexKu(25)
  CHARACTER MSSurfPeakIndexKa(25)
  CHARACTER MSThroughsurfIndexKu(25)
  CHARACTER MSThroughsurfIndexKa(25)
  CHARACTER MSKneeDFRindex(25)
  CHARACTER MSThrZindex(25)
  CHARACTER NUBFratioPIAindex(25)
  CHARACTER NUBFnZmVarIndex(3,25)
  CHARACTER NUBFnZkVarIndex(3,25)
  INTEGER*2 NUBFnZmVarScaling(25)
INTEGER*2 NUBFnZkVarScaling(25)
REAL*4 NUBFsurfSliceIndex(30,25)
REAL*4 NUBFprofZPC(30,25)
INTEGER*2 MSbreakpoints(13,25)
REAL*4 MSslopes(10,25)
REAL*4 MSslopePoints(13,25)
REAL*4 MSslopeFits(6,25)
CHARACTER MSlowSNRrangeFilter(4,25)
REAL*4 NUBFcorrPIA(2,25)
REAL*4 triggerParameters(8,25)
END STRUCTURE

STRUCTURE /L2ADPR_MS_FLG/
  BYTE flagEcho(176,25)
  INTEGER*4 qualityData(25)
  BYTE qualityFlag(25)
  BYTE flagSensor
  INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2ADPR_MS_SLV/
  INTEGER*2 binEchoBottom(25)
  REAL*4 piaFinal(25)
  REAL*4 sigmaZeroCorrected(25)
  REAL*4 zFactorCorrected(176,25)
  REAL*4 zFactorCorrectedESurface(25)
  REAL*4 zFactorCorrectedNearSurface(25)
  REAL*4 paramNUBF(3,25)
  REAL*4 precipWaterIntegrated(2,25)
  REAL*4 precipRateNearSurface(25)
  REAL*4 precipRateESurface(25)
  REAL*4 precipRateAve24(25)
  CHARACTER phaseNearSurface(25)
  REAL*4 epsilon(176,25)
END STRUCTURE

STRUCTURE /L2ADPR_MS_EXPERIMENTAL/
  REAL*4 precipRateESurface2(25)
  CHARACTER precipRateESurface2Status(25)
  REAL*4 sigmaZeroProfile(7,25)
  INTEGER*2 binDEML2(25)
  REAL*4 seaIceConcentration(25)
  CHARACTER flagSurfaceSnowfall(25)
REAL*4 surfaceSnowfallIndex(25)
END STRUCTURE

STRUCTURE /L2ADPR_MS_DSD/
    INTEGER*2 binNode(5,25)
END STRUCTURE

STRUCTURE /L2ADPR_MS_SRT/
    REAL*4 PIAalt(6,25)
    REAL*4 RFactorAlt(6,25)
    REAL*4 PIAweight(6,25)
    REAL*4 pathAtten(25)
    REAL*4 reliabFactor(25)
    INTEGER*2 reliabFlag(25)
    INTEGER*2 refScanID(2,2,25)
    REAL*4 PIAhb(25)
    REAL*4 PIAhybrid(25)
    REAL*4 zeta(25)
    REAL*4 stddevEff(3,25)
    REAL*4 reliabFactorHY(25)
    REAL*4 stddevHY(25)
    INTEGER*2 reliabFlagHY(25)
END STRUCTURE

STRUCTURE /L2ADPR_MS_CSF/
    INTEGER*4 flagBB(25)
    INTEGER*2 binBBPeak(25)
    INTEGER*2 binBBTop(25)
    INTEGER*2 binDFRmMLBottom(25)
    INTEGER*2 binDFRmMLTop(25)
    INTEGER*2 binBBBottom(25)
    REAL*4 heightBB(25)
    REAL*4 widthBB(25)
    INTEGER*4 qualityBB(25)
    INTEGER*4 typePrecip(25)
    INTEGER*4 qualityTypePrecip(25)
    INTEGER*4 flagShallowRain(25)
    BYTE flagHeavyIcePrecip(25)
END STRUCTURE

STRUCTURE /L2ADPR_MS_VER/
    INTEGER*2 binZeroDeg(25)
    REAL*4 attenuationNP(176,25)
REAL*4 piaNP(4,25)
REAL*4 sigmaZeroNPCorrected(25)
REAL*4 heightZeroDeg(25)
END STRUCTURE

STRUCTURE /L2ADPR_MS_PRE/
  REAL*4 elevation(25)
  INTEGER*4 landSurfaceType(25)
  REAL*4 localZenithAngle(25)
  INTEGER*4 flagPrecip(25)
  CHARACTER flagSigmaZeroSaturation(25)
  INTEGER*2 binRealSurface(25)
  INTEGER*2 binStormTop(25)
  REAL*4 heightStormTop(25)
  INTEGER*2 binClutterFreeBottom(25)
  REAL*4 sigmaZeroMeasured(25)
  REAL*4 zFactorMeasured(176,25)
  REAL*4 ellipsoidBinOffset(25)
  REAL*4 snRatioAtRealSurface(25)
  REAL*4 adjustFactor(25)
  BYTE snowIceCover(25)
END STRUCTURE

STRUCTURE /L2ADPR_MS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2ADPR_MS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(25)
  REAL*4 Longitude(25)
RECORD /L2ADPR_MS_SCANSTATUS/ scanStatus
RECORD /NAVIGATION/ navigation
RECORD /L2ADPR_MS_PRE/ PRE
RECORD /L2ADPR_MS_VER/ VER
RECORD /L2ADPR_MS_CSF/ CSF
RECORD /L2ADPR_MS_SRT/ SRT
RECORD /L2ADPR_MS_DSD/ DSD
RECORD /L2ADPR_MS_EXPERIMENTAL/ Experimental
RECORD /L2ADPR_MS_SLV/ SLV
RECORD /L2ADPR_MS_FLG/ FLG
RECORD /L2ADPR_MS_TRG/ TRG
END STRUCTURE

STRUCTURE /L2ADPR_NS_FLG/
  BYTE flagEcho(176,49)
  INTEGER*4 qualityData(49)
  BYTE qualityFlag(49)
  BYTE flagSensor
  INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2ADPR_NS_SLV/
  BYTE flagSLV(176,49)
  REAL*4 paramDSD(2,176,49)
  INTEGER*2 binEchoBottom(49)
  REAL*4 piaFinal(49)
  REAL*4 sigmaZeroCorrected(49)
  REAL*4 zFactorCorrected(176,49)
  REAL*4 zFactorCorrectedESurface(49)
  REAL*4 zFactorCorrectedNearSurface(49)
  REAL*4 paramNUBF(3,49)
  REAL*4 precipRate(176,49)
  REAL*4 precipWaterIntegrated(2,49)
  INTEGER*4 qualitySLV(49)
  REAL*4 precipRateNearSurface(49)
  REAL*4 precipRateESurface(49)
  REAL*4 precipRateAve24(49)
  CHARACTER phaseNearSurface(49)
  REAL*4 epsilon(176,49)
END STRUCTURE

STRUCTURE /L2ADPR_NS_EXPERIMENTAL/
  REAL*4 precipRateESurface2(49)
CHARACTER precipRateESurface2Status(49)
REAL*4 sigmaZeroProfile(7,49)
INTEGER*2 binDEML2(49)
REAL*4 seaIceConcentration(49)
END STRUCTURE

STRUCTURE /L2ADPR_NS_DSD/
  CHARACTER phase(176,49)
  INTEGER*2 binNode(5,49)
END STRUCTURE

STRUCTURE /L2ADPR_NS_SRT/
  REAL*4 PIAalt(6,49)
  REAL*4 RFactorAlt(6,49)
  REAL*4 PIAweight(6,49)
  REAL*4 pathAtten(49)
  REAL*4 reliabFactor(49)
  INTEGER*2 reliabFlag(49)
  INTEGER*2 refScanID(2,2,49)
  REAL*4 PIAb(49)
  REAL*4 PIAhybrid(49)
  REAL*4 zeta(49)
  REAL*4 stddevEff(3,49)
  REAL*4 reliabFactorHY(49)
  REAL*4 stddevHY(49)
  INTEGER*2 reliabFlagHY(49)
END STRUCTURE

STRUCTURE /L2ADPR_NS_CSF/
  INTEGER*4 flagBB(49)
  INTEGER*2 binBBPeak(49)
  INTEGER*2 binBBTop(49)
  INTEGER*2 binBBBottom(49)
  REAL*4 heightBB(49)
  REAL*4 widthBB(49)
  INTEGER*4 qualityBB(49)
  INTEGER*4 typePrecip(49)
  INTEGER*4 qualityTypePrecip(49)
  INTEGER*4 flagShallowRain(49)
  BYTE flagHeavyIcePrecip(49)
  BYTE flagAnvil(49)
END STRUCTURE
STRUCTURE /L2ADPR_NS_VER/
   INTEGER*2 binZeroDeg(49)
   REAL*4 attenuationNP(176,49)
   REAL*4 piaNP(4,49)
   REAL*4 sigmaZeroNPCorrected(49)
   REAL*4 heightZeroDeg(49)
END STRUCTURE

STRUCTURE /L2ADPR_NS_PRE/
   REAL*4 elevation(49)
   INTEGER*4 landSurfaceType(49)
   REAL*4 localZenithAngle(49)
   INTEGER*4 flagPrecip(49)
   CHARACTER flagSigmaZeroSaturation(49)
   INTEGER*2 binRealSurface(49)
   INTEGER*2 binStormTop(49)
   REAL*4 heightStormTop(49)
   INTEGER*2 binClutterFreeBottom(49)
   REAL*4 sigmaZeroMeasured(49)
   REAL*4 zFactorMeasured(176,49)
   REAL*4 ellipsoidBinOffset(49)
   REAL*4 snRatioAtRealSurface(49)
   REAL*4 adjustFactor(49)
   BYTE snowIceCover(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
   REAL*4 scPos(3)
   REAL*4 scVel(3)
   REAL*4 scLat
   REAL*4 scLon
   REAL*4 scAlt
   REAL*4 dprAlt
   REAL*4 scAttRollGeoc
   REAL*4 scAttPitchGeoc
   REAL*4 scAttYawGeoc
   REAL*4 scAttRollGeod
   REAL*4 scAttPitchGeod
   REAL*4 scAttYawGeod
   REAL*4 greenHourAng
   REAL*8 timeMidScan
   REAL*8 timeMidScanOffset
END STRUCTURE
STRUCTURE /L2ADPR_NS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2ADPR_NS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
  REAL*4 Longitude(49)
  RECORD /L2ADPR_NS_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L2ADPR_NS_PRE/ PRE
  RECORD /L2ADPR_NS_VER/ VER
  RECORD /L2ADPR_NS_CSF/ CSF
  RECORD /L2ADPR_NS_SRT/ SRT
  RECORD /L2ADPR_NS_DSD/ DSD
  RECORD /L2ADPR_NS_EXPERIMENTAL/ Experimental
  RECORD /L2ADPR_NS_SLV/ SLV
  RECORD /L2ADPR_NS_FLG/ FLG
END STRUCTURE

STRUCTURE /L2ADPR_SWATHS/
   RECORD /L2ADPR_NS/ NS;
   RECORD /L2ADPR_MS/ MS;
   RECORD /L2ADPR_HS/ HS;
END STRUCTURE

5.51 2APR - PR precipitation

The PR Level-2A product, 2APR, "PR precipitation," is written as a 1 swath structure. The swath is NS, normal scans. The following sections describe the structure and contents of the format.

Dimension definitions:

- nscan: Number of scans in the granule.
- nray: Number of angle bins in each NS scan.
- nrayMS: Number of angle bins in each MS scan.
- nrayHS: Number of angle bins in each HS scan.
- nbin: Number of range bins in each NS and MS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
- nbinSZP: Number of range bins for sigmaZeroProfile.
- nNP: Number of NP kinds.
- foreBack: Forward, Backward.
- method: Number of SRT methods.
- nsdew: Number of standard deviation effective ways.
- nNode: Number of binNode.
- nDSD: Number of DSD parameters. Parameters are dBNw and Dm (mm).
- LS: Liquid, solid.
- nNUBF: Number of NUBF parameters.

Figure 589 through Figure 600 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 589: Data Format Structure for 2APR, PR precipitation
Figure 590: Data Format Structure for 2APR, ScanTime

Figure 591: Data Format Structure for 2APR, scanStatus
Figure 592: Data Format Structure for 2APR, navigation
Figure 593: Data Format Structure for 2APR, PRE

Figure 594: Data Format Structure for 2APR, VER
5.51 2APR - PR precipitation

Figure 595: Data Format Structure for 2APR, CSF

Figure 596: Data Format Structure for 2APR, SRT
Figure 597: Data Format Structure for 2APR, DSD

Figure 598: Data Format Structure for 2APR, Experimental
5.51 2APR - PR precipitation

![Diagram of data format structure for 2APR, SLV]

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

NS (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scanStatus (Group)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

Bit  Meaning if bit = 1
0  missing
5  geoError is not zero
6  modeStatus is not zero
**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorrientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$ the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$ the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
</tbody>
</table>
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
</tbody>
</table>
2. **SUNPOINT**
3. **GSPM** (Gyro-less Sun Point)
4. **MSM** (Mission Science Mode)
5. **SLEW**
6. **DELTAH**
7. **DELTAV**

-99 **UNKNOWN -- ACS mode unavailable**

**targetSelectionMidScan** (1-byte integer, array size: nscan): targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
</tbody>
</table>
17  Ku/Ka Independent Standby VPRF Table OUT
18  Ku/Ka Independent Standby Phase Out
19  Ku/Ka Independent Standby Dump Out
20  Ku/Ka Independent Standby (No Science Data)

limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks.
limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPos (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Inertial (ECI) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector (ms$^{-1}$) of the spacecraft in ECI Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
- -9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
- -9999.9 Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
- -9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
- -9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
- -9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**PRE** (Group)

**elevation** (4-byte float, array size: nray x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:
-9999.9 Missing value

**landSurfaceType** (4-byte integer, array size: nray x nscan):

Land surface type.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 99</td>
<td>Ocean</td>
</tr>
<tr>
<td>100 - 199</td>
<td>Land</td>
</tr>
<tr>
<td>200 - 299</td>
<td>Coast</td>
</tr>
<tr>
<td>300 - 399</td>
<td>Inland water</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing value</td>
</tr>
</tbody>
</table>
localZenithAngle (4-byte float, array size: nray x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values are in degree. Special values are defined as:
-9999.9 Missing value

flagPrecip (4-byte integer, array size: nray x nscan):
Precipitation or no precipitation.

For L2 Ku and L2 Ka

0 No precipitation
1 Precipitation
-9999 Missing value

For L2 DPR

0 No precipitation by both Ku and Ka
1 Precipitation by Ka, no rain by Ku
10 Precipitation by Ku, no rain by Ka
11 Precipitation by both Ku and Ka
-9999 Missing value

flagSigmaZeroSaturation (1-byte char, array size: nray x nscan):
A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nray x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nray x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based
ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

**heightStormTop** (4-byte float, array size: nray x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value

**binClutterFreeBottom** (2-byte integer, array size: nray x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

**sigmaZeroMeasured** (4-byte float, array size: nray x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:
-9999.9 Missing value

**zFactorMeasured** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**ellipsoidBinOffset** (4-byte float, array size: nray x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

\[
\text{ellipsoidBinOffset} = \text{scRangeEllipsoid} - \{ \text{startBinRange} + (\text{binEllipsoid}-1) \times \text{rangeBinSize} \}
\]

\begin{align*}
\text{scRangeEllipsoid} & : \text{Distance between a sensor and the ellipsoid [m]} \\
\text{startBinRange} & : \text{Distance between a sensor and a center of the highest observed range bin [m]} \\
\text{binEllipsoid} & : \text{Range bin number of the Ellipsoid (1 - 260)} \\
\text{rangeBinSize} & : \text{Range bin size [m]}
\end{align*}

-9999 Missing value

**snRatioAtRealSurface** (4-byte float, array size: nray x nscan):
Signal/Noise ratio at real surface range bin.

\[
\text{snRatioAtRealSurface} = 10 \times \log_{10}(\text{echoPowertrueV[mW]}/\text{noisePowertrueV[mW]})
\]

-9999 Missing value
adjustFactor (4-byte float, array size: nray x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm’) and sigmaZeroMeasured (dBs0m’).

dBZm’ = dBZm - adjustFactor

dBs0m’ = dBs0m - adjustFactor

The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
temporal adjustment for orbit number dependency.

snowIceCover (1-byte integer, array size: nray x nscan):
TBD. Special values are defined as:
-99 Missing value

VER (Group)

binZeroDeg (2-byte integer, array size: nray x nscan):

Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

attenuationNP (4-byte float, array size: nbin x nray x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud
ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are
defined as:
-9999.9 Missing value

piaNP (4-byte float, array size: nNP x nray x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water,
cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are
defined as:
-9999.9 Missing value

**sigmaZeroNPCorrected** (4-byte float, array size: nray x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

**heightZeroDeg** (4-byte float, array size: nray x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9 Missing value

**CSF** (Group)

**flagBB** (4-byte integer, array size: nray x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.

**L2 DPR:**
0  no Bright Band
1  Bright Band detected by Ku and DFRm
2  Bright Band detected by Ku only
3  Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value

**L2 Ku and L2 Ka:**
0  BB not detected
1  BB detected
-1111 No rain value
-9999 Missing value

**binBBPeak** (2-byte integer, array size: nray x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binBBTop** (2-byte integer, array size: nray x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88
at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binBBBottom** (2-byte integer, array size: nray x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**heightBB** (4-byte float, array size: nray x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

**widthBB** (4-byte float, array size: nray x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

**qualityBB** (4-byte integer, array size: nray x nscan):

Quality of the bright band.
When the bright band is detected,
a larger positive number indicates lower confidence in the detection.

The Ku detection is clear, but the Ka and DPR detection is somewhat doubtful.

The meaning of qualityBB has not been finalized.

3 Smeared bright band
2 Not so clear bright band
1 Clear bright band
0 BB not detected in the case of rain
-1111 No rain value
-9999 Missing value

**typePrecip** (4-byte integer, array size: nray x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, onvective, and other, can be obtained as follows:
When typePrecip is greater than zero,
Major rain type $= \frac{\text{typePrecip}}{10000000}$

- 1  stratiform
- 2  convective
- 3  other

-1111 No rain value
-9999 Missing value

Let abcdefgh be the 8 digit number,

\[
\begin{align*}
\text{abcdefgh} \\
\text{then} \\
\text{a: Main rain type. (a=1,2,3),} \\
\text{b: 0,} \\
\text{c: 0,} \\
\text{d: V rain type,} \\
\text{e: H rain type,} \\
\text{f: BB,} \\
\text{g: Shallow rain,} \\
\text{h: Small size cell.}
\end{align*}
\]

The following numbers appear as Ku and Ka (MS/HS) rain types:

--- stratiform
1001H100
10031000
--- convective
2001H1xy (x>0 or y>0)
2002Hbxy
200310xy (x>0 or y>0)
200320xy
--- other
300330xy

where $H$ is the rain type by H-method, and $b$ depends on BB, $x$ on shallow rain and $y$ on small size cell:

- $H = 1$: stratiform by H-method,
- 2: convective by H-method,
- 3: other by H-method.

- $b = 0$: BB not detected,
- 1: BB detected.
x = 0: No shallow rain,
    1: Shallow isolated,
    3: Shallow non-isolated.

y = 0: No small size cell,
    1: Single cell,
    2: Small size cell consisting of two adjacent pixels.

In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:

DFRm rain type = (typePrecip%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000 in FORTRAN

DFRm rain type
    = 1   stratiform
    = 2   convective
    = 4   transition
    = 8   DFRm method cannot be applicable at Part B (in this case the conventional method determines the major rain type)
    = 9   DFRm method cannot be applicable at Part A (in this case the conventional method determines the major rain type)

-1111  No rain value
-9999   Missing value

If dual frequency data is not available but Ku-only or Ka-only is available, rain type is expressed by the following 8 digit number:
    10xxxxxx --- stratiform,
    20xxxxxx --- convective,
    30xxxxxx --- other,
which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by
    1qxxxxxx --- stratiform,
    2qxxxxxx --- convective,
    3qxxxxxx --- other,
where q>0.
Thus, by examining q, users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.
For MS and HS, DFRm method is used.

DFRm decision classifies rain type into
stratiform,
convective,
and
transition.

The DPR numbering rule can be summarized as follows:

Let opqrstuv be the 8 digit number, then

\( o \): Main rain type. \((o=1,2,3)\),
\( p \): DFRm rain type. \((p=0,1,2,4,8,9, \text{with } p=0 \text{ for single frequency data only})\),
\( q \): DFRm BB. \((q=0,1)\),
\( r \): V rain type (by conventional V-method).

Basically \( r=0 \) for inner swath and \( r>0 \) for outer swath.
However, \( r>0 \) when only single frequency data is available,
\( s \): H rain type,
\( t \): = 0 for inner swath,
\( 1 \) when BB is detected in the outer swath.
\( u \): Shallow rain,
\( v \): Small size cell.

DFRm type can be obtained by examining \( p \)

The meaning of \( p \) is as follows:

\( p = 0 \): single frequency data only (dual frequency data not available),
\( 1 \): stratiform by DFRm method,
\( 2 \): convective by DFRm method,
\( 4 \): transition by DFRm method,
\( 8 \): DFRm decision not available,
\( 9 \): DFRm decision not available.

Note that \( p>0 \) always in DPR processing, which is different from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, \( p=0 \) always.

The following numbers appear as DPR rain types:
--- stratiform
1901H100
19031000
--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy
--- other
390330xy

*******************************
* For NS inner swath and MS *
*******************************
--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
(x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
(x>0 or y>0)
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type

***************
* For HS *
***************
--- stratiform
11B0H000
14B01000
19001000 --- H decision only
--- convective
21B0H0x0 (x>0)
22B0H0x0
240010x0 (x>0, 24B010x0 with B=0)
240020x0
241010x0 (x>0, 24B010x0 with B=1)
290010x0 (x>0) --- H decision only
290020x0 --- H decision only
--- other
340030x0
390030x0 --- H decision only

where w depends on BB by conventional V-method, B on BB
by DFRm method, H on H-method, x on shallow rain
and y on small size cell:
  w = 0: BB not detected by conventional V-method,
        1: BB detected by conventional V-method.

  B = 0: BB not detected by DFRm method,
        1: BB detected by DFRm method.

  H = 1: stratiform by H-method,
        2: convective by H-method,
        3: other by H-method.

  x = 0: No shallow rain,
        1: Shallow isolated,
        3: Shallow non-isolated.

  y = 0: No small size cell,
        1: Single cell,
2: Small size cell consisting of two adjacent pixels. In the above, x>0 and y>0 are taken care of in the function R\_type\_classification\_dpr2().

qualityTypePrecip (4-byte integer, array size: nray x nscan):

Quality of the precipitation type.

1 Good
-1111 No rain value
-9999 Missing value

flagShallowRain (4-byte integer, array size: nray x nscan):

Type of shallow rain
0 No shallow rain
10 Shallow isolated (maybe)
11 Shallow isolated (certain)
20 Shallow non-isolated (maybe)
21 Shallow non-isolated (certain)
-1111 No rain value
-9999 Missing value

flagHeavyIcePrecip (1-byte integer, array size: nray x nscan):
This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:

-99 Missing value

flagAnvil (1-byte integer, array size: nray x nscan):

flagAnvil is 1 when anvil is detected by the Ku-band radar, 0 when anvil is not detected, and -99 when the data is missing.

Note that Ka-band decision is not made because of a lower sensitivity of Ka-band radar (therefore, there does not exist any Ka-band flagAnvil; only Ku-band flagAnvil is available in Ku-only and DPR NS).
PIAalt (4-byte float, array size: method x nray x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where

- PIAalt (j=1) = PIA_Ku from forward along-track spatial at kth angle bin
- PIAalt (j=2) = PIA_Ku from backward along-track spatial at kth angle bin
- PIAalt (j=3) = PIA_Ku from forward hybrid at kth angle bin
- PIAalt (j=4) = PIA_Ku from backward hybrid at kth angle bin
- PIAalt (j=5) = PIA_Ku from temporal reference at kth angle bin
- PIAalt (j=6) = PIA_Ku from light-rain temporal reference at kth angle bin

Values are in dB. Special values are defined as:
-9999.9 Missing value

RFactorAlt (4-byte float, array size: method x nray x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAalt. Special values are defined as:
-9999.9 Missing value

PIAweight (4-byte float, array size: method x nray x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

PIAweight_j = 1/sigma_j^2 * (1/Sum_j(1/sigma_j^2))

Special values are defined as:
-9999.9 Missing value

pathAtten (4-byte float, array size: nray x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
-9999.9 Missing value

reliabFactor (4-byte float, array size: nray x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
-9999.9 Missing value

reliabFlag (2-byte integer, array size: nray x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:

-1 if Rel_eff > 3 ; PIAeff estimate is considered reliable
-2 if 3 ≥ Rel_eff > 1 ; PIAeff estimate is considered marginally reliable
-3 if Rel_eff ≤ 1 ; PIAeff is unreliable
-4 if SNR_at surface < 2dB; provides a lower bound to the path-attenuation
-9 (no-rain case)
Special values are defined as:
   -9999   Missing value

**refScanID** (2-byte integer, array size: nearFar x foreBack x nray x nscan):
The number of scan lines between the current scan and the beginning (or end) of the
along-track reference data at each angle bin. The values are computed by the equation:
Current Scan Number - Reference Scan Number. The values are positive for the Forward
estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:

1,1 - Forward - Near reference
2,1 - Forward - Far reference
1,2 - Backward - Near reference
2,2 - Backward - Far reference

Special values are defined as:
   -9999   Missing value

**PIAhb** (4-byte float, array size: nray x nscan):
The 2-way attenuation of HB.

Values are in dB. Special values are defined as:
   -9999.9   Missing value

**PIAhybrid** (4-byte float, array size: nray x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as:
   -9999.9   Missing value

**zeta** (4-byte float, array size: nray x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
   -9999.9   Missing value

**stddevEff** (4-byte float, array size: nsdew x nray x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.
Special values are defined as:
-9999.9 Missing value

**reliabFactorHY** (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

**stddevHY** (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

**reliabFlagHY** (2-byte integer, array size: nray x nscan):
TBD.

Special values are defined as:
-9999 Missing value

**DSD (Group)**

**phase** (1-byte char, array size: nbin x nray x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

- phase < 100 Temperature(C)=phase-100
- phase > 200 Temperature(C)=phase-200
- phase = 100 Top of the bright band
- phase = 200 Bottom of the bright band
- phase = 125 is used for the range bins between
  the top and peak of bright band
- phase = 175 is used for the range bins between
  the peak and bottom of bright band

Integer values of phase/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

\textit{binNode} (2-byte integer, array size: nNode x nray x nscan):

The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to
   \textit{binRealSurface} in PRE group.

For NS and MS swaths, bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths, bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
-9999 - Missing

\textbf{Experimental (Group)}

\textit{precipRateESurface2} (4-byte float, array size: nray x nscan):
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

\textit{precipRateESurface2Status} (1-byte char, array size: nray x nscan):
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
255 Missing value

\textit{sigmaZeroProfile} (4-byte float, array size: nbinSZP x nray x nscan):
Surface backscattering cross section profile around the current ifov. For information on
this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:
-9999.9 Missing value

binDEML2 (2-byte integer, array size: nray x nscan):
Range bin number of the digital elevation model surface estimate. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
-9999 Missing value

seaIceConcentration (4-byte float, array size: nray x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value

SLV (Group)

flagSLV (1-byte integer, array size: nbin x nray x nscan):
Special values are defined as:
-99 Missing value

paramDSD (4-byte float, array size: nDSD x nbin x nray x nscan):
Parameters of the drop size distribution. The first index is dBNw; the second index is Dm in mm. Special values are defined as:
-9999.9 Missing value

binEchoBottom (2-byte integer, array size: nray x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

piaFinal (4-byte float, array size: nray x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

sigmaZeroCorrected (4-byte float, array size: nray x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9 Missing value

zFactorCorrected (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9 Missing value
**zFactorCorrectedESurface** (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**paramNUBF** (4-byte float, array size: nNUBF x nray x nscan):
TBD. Special values are defined as:
-9999.9 Missing value

**precipRate** (4-byte float, array size: nbin x nray x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipWaterIntegrated** (4-byte float, array size: LS x nray x nscan):
Precipitation water vertically integrated. Values are in g/m². Special values are defined as:
-9999.9 Missing value

**qualitySLV** (4-byte integer, array size: nray x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are defined as:
-9999 Missing value

**precipRateNearSurface** (4-byte float, array size: nray x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface** (4-byte float, array size: nray x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateAve24** (4-byte float, array size: nray x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**phaseNearSurface** (1-byte char, array size: nray x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

phaseNearSurface < 100 Temperature(C)=phaseNearSurface-100
phaseNearSurface > 200 Temperature(C)=phaseNearSurface-200
phaseNearSurface = 100 Top of the bright band
phaseNearSurface = 200 Bottom of the bright band
phaseNearSurface = 125 is used for the range bins between
the top and peak of bright band
phaseNearSurface = 175 is used for the range bins between
the peak and bottom of bright band

Integer values of phaseNearSurface/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

epsilon (4-byte float, array size: nbin x nray x nscan): Epsilon is the indication of the adjustment away from the initial drop size distribution, epsilon = 1 is no adjustment. Special values are defined as:
-9999.9 Missing value

FLG (Group)

flagEcho (1-byte integer, array size: nbin x nray x nscan):

Flag of precipitation and main/side lobe clutter information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>

qualityData (4-byte integer, array size: nray x nscan): Normal data gives ”0”. Non-zero values mean the kinds of errors. Special values are defined as:
-9999 Missing value
Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag has 2 bits of information.

The 2 bit flag for each module has values:

- [0 0] Good
- [0 1] Warning but usable
- [1 0] NG or error

The bits of qualityData are assigned as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 7</td>
<td>Copy of dataQuality in level 1B product</td>
</tr>
<tr>
<td>8 - 9</td>
<td>Flag by input module</td>
</tr>
<tr>
<td>10 - 11</td>
<td>Flag by preparation module</td>
</tr>
<tr>
<td>12 - 13</td>
<td>Flag by vertical module</td>
</tr>
<tr>
<td>14 - 15</td>
<td>Flag by classification module</td>
</tr>
<tr>
<td>16 - 17</td>
<td>Flag by SRT module</td>
</tr>
<tr>
<td>18 - 19</td>
<td>Flag by DSD module</td>
</tr>
<tr>
<td>20 - 21</td>
<td>Flag by solver module</td>
</tr>
<tr>
<td>22 - 23</td>
<td>Flag by output module</td>
</tr>
<tr>
<td>24 - 31</td>
<td>Spare</td>
</tr>
</tbody>
</table>

**qualityFlag** (1-byte integer, array size: nray x nscan):
Flag derived from qualityData with the following values: Special values are defined as:

-99 Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

**flagSensor** (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>
flagScanPattern (2-byte integer, array size: nscan):

Flag of scan pattern.

Value  Meaning
1  TBD
-9999  Missing

C Structure Header file:

```c
#ifndef _TK_2APR_H_
#define _TK_2APR_H_

#ifndef _L2APR_FLG_
#define _L2APR_FLG_

typedef struct {
    signed char flagEcho[49][176];
    int qualityData[49];
    signed char qualityFlag[49];
    signed char flagSensor;
    short flagScanPattern;
} L2APR_FLG;
#endif

#ifndef _L2APR_SLV_
#define _L2APR_SLV_

typedef struct {
    signed char flagSLV[49][176];
    float paramDSD[49][176][2];
    short binEchoBottom[49];
    float piaFinal[49];
    float sigmaZeroCorrected[49];
    float zFactorCorrected[49][176];
    float zFactorCorrectedESurface[49];
    float zFactorCorrectedNearSurface[49];
    float paramNUBF[49][3];
    float precipRate[49][176];
    float precipWaterIntegrated[49][2];
    int qualitySLV[49];
} L2APR_SLV;
#endif
```
float precipRateNearSurface[49];
float precipRateESurface[49];
float precipRateAve24[49];
unsigned char phaseNearSurface[49];
float epsilon[49][176];
}
L2APR_SLV;

#endif
#ifndef _L2APR_EXPERIMENTAL_
#define _L2APR_EXPERIMENTAL_

typedef struct {
    float precipRateESurface2[49];
    unsigned char precipRateESurface2Status[49];
    float sigmaZeroProfile[49][7];
    short binDEML2[49];
    float seaIceConcentration[49];
} L2APR_EXPERIMENTAL;

#endif
#ifndef _L2APR_DSD_
#define _L2APR_DSD_

typedef struct {
    unsigned char phase[49][176];
    short binNode[49][5];
} L2APR_DSD;

#endif
#ifndef _L2APR_SRT_
#define _L2APR_SRT_

typedef struct {
    float PIAalt[49][6];
    float RFactorAlt[49][6];
    float PIAweight[49][6];
    float pathAtten[49];
    float reliabFactor[49];
    short reliabFlag[49];
    short refScanID[49][2][2];
}
float PIAhb[49];
float PIAhybrid[49];
float zeta[49];
float stddevEff[49][3];
float reliabFactorHY[49];
float stddevHY[49];
short reliabFlagHY[49];
} L2APR_SRT;

#endif

#ifndef _L2APR_CSF_
#define _L2APR_CSF_

typedef struct {
  int flagBB[49];
  short binBBPeak[49];
  short binBBTop[49];
  short binBBBottom[49];
  float heightBB[49];
  float widthBB[49];
  int qualityBB[49];
  int typePrecip[49];
  int qualityTypePrecip[49];
  int flagShallowRain[49];
  signed char flagHeavyIcePrecip[49];
  signed char flagAnvil[49];
} L2APR_CSF;

#endif

#ifndef _L2APR_VER_
#define _L2APR_VER_

typedef struct {
  short binZeroDeg[49];
  float attenuationNP[49][176];
  float piaNP[49][4];
  float sigmaZeroNPCorrected[49];
  float heightZeroDeg[49];
} L2APR_VER;

#endif
#ifndef _L2APR_PRE_
#define _L2APR_PRE_

typedef struct {
    float elevation[49];
    int landSurfaceType[49];
    float localZenithAngle[49];
    int flagPrecip[49];
    unsigned char flagSigmaZeroSaturation[49];
    short binRealSurface[49];
    short binStormTop[49];
    float heightStormTop[49];
    short binClutterFreeBottom[49];
    float sigmaZeroMeasured[49];
    float zFactorMeasured[49][176];
    float ellipsoidMeasured[49][176];
    float snRatioAtRealSurface[49];
    float adjustFactor[49];
    signed char snowIceCover[49];
} L2APR_PRE;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} NAVIGATION;

#endif
typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2APR_SCANSTATUS;

 ifndef  _L2APR_SCANSTATUS_
 #define  _L2APR_SCANSTATUS_

 typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
 } SCANTIME;

 ifndef  _SCANTIME_
 #define  _SCANTIME_

 typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
 } SCANTIME;

 ifndef  _L2APR_NS_
 #define  _L2APR_NS_

 endif

 ifndef  _L2APR_SCANSTATUS_
 #define  _L2APR_SCANSTATUS_

 typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2APR_SCANSTATUS;

 endif

 ifndef  _SCANTIME_
 #define  _SCANTIME_

 typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
 } SCANTIME;

 endif

 ifndef  _L2APR_NS_
 #define  _L2APR_NS_
#define _L2APR_NS_

typedef struct {
   SCANTIME ScanTime;
   float Latitude[49];
   float Longitude[49];
   L2APR_SCANSTATUS scanStatus;
   NAVIGATION navigation;
   L2APR_PRE PRE;
   L2APR_VER VER;
   L2APR_CSF CSF;
   L2APR_SRT SRT;
   L2APR_DSD DSD;
   L2APR_EXPERIMENTAL Experimental;
   L2APR_SLV SLV;
   L2APR_FLG FLG;
} L2APR_NS;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L2APR_FLG/
   BYTE flagEcho(176,49)
   INTEGER*4 qualityData(49)
   BYTE qualityFlag(49)
   BYTE flagSensor
   INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2APR_SLV/
   BYTE flagSLV(176,49)
   REAL*4 paramDSD(2,176,49)
   INTEGER*2 binEchoBottom(49)
   REAL*4 piaFinal(49)
   REAL*4 sigmaZeroCorrected(49)
   REAL*4 zFactorCorrected(176,49)
   REAL*4 zFactorCorrectedESurface(49)
   REAL*4 zFactorCorrectedNearSurface(49)
   REAL*4 paramNUBF(3,49)
   REAL*4 precipRate(176,49)
REAL*4 precipWaterIntegrated(2,49)
INTEGER*4 qualitySLV(49)
REAL*4 precipRateNearSurface(49)
REAL*4 precipRateESurface(49)
REAL*4 precipRateAve24(49)
CHARACTER phaseNearSurface(49)
REAL*4 epsilon(176,49)

END STRUCTURE

STRUCTURE /L2APR_EXPERIMENTAL/
  REAL*4 precipRateESurface2(49)
  CHARACTER precipRateESurface2Status(49)
  REAL*4 sigmaZeroProfile(7,49)
  INTEGER*2 binDEML2(49)
  REAL*4 seaIceConcentration(49)
END STRUCTURE

STRUCTURE /L2APR_DSD/
  CHARACTER phase(176,49)
  INTEGER*2 binNode(5,49)
END STRUCTURE

STRUCTURE /L2APR_SRT/
  REAL*4 PIAalt(6,49)
  REAL*4 RFactorAlt(6,49)
  REAL*4 PIAweight(6,49)
  REAL*4 pathAtten(49)
  REAL*4 reliabFactor(49)
  INTEGER*2 reliabFlag(49)
  INTEGER*2 refScanID(2,2,49)
  REAL*4 PIAhb(49)
  REAL*4 PIAhybrid(49)
  REAL*4 zeta(49)
  REAL*4 stddevEff(3,49)
  REAL*4 reliabFactorHY(49)
  REAL*4 stddevHY(49)
  INTEGER*2 reliabFlagHY(49)
END STRUCTURE

STRUCTURE /L2APR_CSF/
  INTEGER*4 flagBB(49)
  INTEGER*2 binBBPeak(49)
  INTEGER*2 binBBTop(49)
INTEGER*2 binBBBottom(49)
REAL*4 heightBB(49)
REAL*4 widthBB(49)
INTEGER*4 qualityBB(49)
INTEGER*4 typePrecip(49)
INTEGER*4 qualityTypePrecip(49)
INTEGER*4 flagShallowRain(49)
BYTE flagHeavyIcePrecip(49)
BYTE flagAnvil(49)
END STRUCTURE

STRUCTURE /L2APR_VER/
    INTEGER*2 binZeroDeg(49)
    REAL*4 attenuationNP(176,49)
    REAL*4 piaNP(4,49)
    REAL*4 sigmaZeroNPCorrected(49)
    REAL*4 heightZeroDeg(49)
END STRUCTURE

STRUCTURE /L2APR_PRE/
    REAL*4 elevation(49)
    INTEGER*4 landSurfaceType(49)
    REAL*4 localZenithAngle(49)
    INTEGER*4 flagPrecip(49)
    CHARACTER flagSigmaZeroSaturation(49)
    INTEGER*2 binRealSurface(49)
    INTEGER*2 binStormTop(49)
    REAL*4 heightStormTop(49)
    INTEGER*2 binClutterFreeBottom(49)
    REAL*4 sigmaZeroMeasured(49)
    REAL*4 zFactorMeasured(176,49)
    REAL*4 ellipsoidBinOffset(49)
    REAL*4 snRatioAtRealSurface(49)
    REAL*4 adjustFactor(49)
    BYTE snowIceCover(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
    REAL*4 scPos(3)
    REAL*4 scVel(3)
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L2APR_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2APR_NS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
5.52 3DPR - DPR Full Product

3DPR, "DPR Full Product", computes statistics of the DPR measurements at both a low horizontal resolution (G1, 5° x 5° latitude/longitude) and a high horizontal resolution (G2, 0.25° x 0.25° latitude/longitude). The product can be monthly or daily.

Histograms have the following category thresholds, where
histbin(i) = cat(i) less than x less than or equal to cat(i+1)

- **cat rain** = [ 0.01, ! mm/h (logarithmic steps)
  0.10, 0.13, 0.17, 0.23, 0.30, 0.40,
  0.52, 0.69, 0.91, 1.20, 1.58, 2.08,
  2.75, 3.62, 4.77, 6.29, 8.29, 10.92,
  14.40, 18.97, 25.00, 32.95, 43.43, 57.24,
  75.44, 99.43, 131.04, 172.71, 227.63, 300.00 ],

- **cat Z** = [ 0.01, ! dBZ
  6.0, 8.0, 10.0, 12.0, 14.0, 16.0,
  18.0, 20.0, 22.0, 24.0, 26.0, 28.0,
  30.0, 32.0, 34.0, 36.0, 38.0, 40.0,
  42.0, 44.0, 46.0, 48.0, 50.0, 52.0,
  54.0, 56.0, 58.0, 60.0, 62.0, 64.0 ],

- **cat integratedWater** = [ 0.0, ! kg/m^2
  200.0, 400.0, 600.0, 800.0, 1000.0, 1200.0,
  1400.0, 1600.0, 1800.0, 2000.0, 2200.0, 2400.0,
  2600.0, 2800.0, 3000.0, 3200.0, 3400.0, 3600.0,
  3800.0, 4000.0, 4200.0, 4400.0, 4600.0, 4800.0,
  5000.0, 5200.0, 5400.0, 5600.0, 5800.0, 6000.0 ],
cat bbhgt = [ 10.0, ! meters
250.0, 500.0, 750.0, 1000.0, 1250.0, 1500.0,
1750.0, 2000.0, 2250.0, 2500.0, 2750.0, 3000.0,
3250.0, 3500.0, 3750.0, 4000.0, 4250.0, 4500.0,
4750.0, 5000.0, 5250.0, 5500.0, 5750.0, 6000.0,
6250.0, 6500.0, 6750.0, 7000.0, 7500.0, 20000.0 ],

cat bbwdth = [ 0.0, ! meters
125.0, 250.0, 375.0, 500.0, 625.0, 750.0,
875.0, 1000.0, 1125.0, 1250.0, 1375.0, 1500.0,
1625.0, 1750.0, 1875.0, 2000.0, 2125.0, 2250.0,
2375.0, 2500.0, 2625.0, 2750.0, 2875.0, 3000.0,
3125.0, 3250.0, 3375.0, 3500.0, 3625.0, 3750.0 ],

cat stormh = 1000.0*[ 0.01, ! km (convert m > km)
0.5, 1.0, 1.5, 2.0, 2.5, 3.0,
3.5, 4.0, 4.5, 5.0, 5.5, 6.0,
6.5, 7.0, 7.5, 8.0, 8.5, 9.0,
9.5, 10.0, 10.5, 11.0, 11.5, 12.0,
12.5, 13.0, 14.0, 15.0, 16.0, 20.0 ],

cat epsilon = [ 0.0,
0.1, 0.2, 0.3, 0.4, 0.5, 0.6,
0.7, 0.8, 0.9, 1.0, 1.1, 1.2,
1.3, 1.4, 1.5, 1.6, 1.7, 1.8,
1.9, 2.0, 2.1, 2.2, 2.3, 2.4,
2.5, 2.6, 2.7, 2.8, 2.9, 3.0 ],

cat nubf = [ 1.0,
1.05, 1.1, 1.15, 1.2, 1.25, 1.3,
1.35, 1.4, 1.45, 1.5, 1.55, 1.6,
1.65, 1.7, 1.75, 1.8, 1.85, 1.9,
1.95, 2.0, 2.1, 2.2, 2.3, 2.4,
2.5, 2.6, 2.7, 2.8, 2.9, 3.0 ],

cat pia = [ 0.01,
0.1, 0.2, 0.3, 0.4, 0.5, 0.6,
0.8, 1.0, 1.2, 1.4, 1.6, 1.8,
2.0, 2.5, 3.0, 3.5, 4.0, 4.5,
5.0, 5.5, 6.0, 7.0, 8.0, 9.0,
10.0, 15.0, 20.0, 25.0, 30.0, 100.0 ],
\[
\text{cat } dBW = [ 0.1, \\
1.0, \ 2.0, \ 4.0, \ 6.0, \ 8.0, \ 10.0, \\
12.0, \ 14.0, \ 16.0, \ 18.0, \ 20.0, \ 22.0, \\
24.0, \ 26.0, \ 28.0, \ 30.0, \ 32.0, \ 34.0, \\
36.0, \ 38.0, \ 40.0, \ 42.0, \ 44.0, \ 46.0, \\
48.0, \ 50.0, \ 52.0, \ 54.0, \ 56.0, \ 60.0 ]
\]

\[
\text{cat } Dm = [ 0.1, \ \text{mm} \\
0.2, \ 0.3, \ 0.4, \ 0.5, \ 0.6, \ 0.7, \\
0.8, \ 0.9, \ 1.0, \ 1.1, \ 1.2, \ 1.3, \\
1.4, \ 1.5, \ 1.6, \ 1.7, \ 1.8, \ 1.9, \\
2.0, \ 2.1, \ 2.2, \ 2.3, \ 2.4, \ 2.5, \\
2.6, \ 2.7, \ 2.8, \ 2.9, \ 3.0, \ 4.0 ]
\]

Dimension definitions:

- **ltL** 28 Number of low resolution 5° grid intervals of latitude from 70°S to 70°N.
- **lnL** 72 Number of low resolution 5° grid intervals of longitude from 180°W to 180°E.
- **ltH** 536 Number of high resolution 0.25° grid intervals of latitude from 67°S to 67°N.
- **lnH** 1440 Number of high resolution 0.25° grid intervals of longitude from 180°W to 180°E.
- **chn** 5 Number of channels: Ku, Ka, KaHS, DPR, KuMS.
- **inst** 4 Number of instruments: Ku, Ka, KaHS, KuMS.
- **hgt** 5 Number of heights above the earth ellipsoid: 2, 4, 6, 10, and 15 km.
- **tim** 24 Number of hours (local time).
- **ang** 7 Number of angles. The meaning of ang is different for each channel. For Ku channel all indeces are used with the meaning 0, 1, 2, ..., 6 = angle bins 24, (20,28), (16,32), (12,36), (8,40), (3,44), and (0,48). For Ka channel 4 indeces are used with the meaning 0, 1, 2, 3 = angle bins 12, (8,16), (4,20), and (0,24). For KaHS channel 4 indeces are used with the meaning 0, 1, 2, 3 = angle bins (11,2), (7,16), (3,20), and (0,23).
- **rt** 3 Number of rain types: stratiform, convective, all.
- **st** 3 Number of surface types: ocean, land, all.
- **bin** 30 Number of bins in histogram. The thresholds are different for different variables. See the introduction to this algorithm.

Figure 601 through Figure 691 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the
Figure 601: Data Format Structure for 3DPR, DPR Full Product

Fortran Structure Header File.
Figure 602: Data Format Structure for 3DPR, G1, G1

continued on next figure

• • •
continued from last figure

- dBnW
- epsilon DPR
- epsilon
- eta
- pia HB
- pia Hybrid
- pia Hybrid DPR
- pia SRT
- pia SRT DPR
- pia Final
- pia Final DPR
- pia Final Subset
- pia Final DPR Subset
- height BB
- height BB Nadir
- BB width Nadir
- height Storm Top
- BB width
- observation Counts
- precip Rate Local Time
- DFR m Near Surface
- DFR Near Surface
- precip Rate Near Surface Unconditional
  - 4 bytes
- precip Probability Near Surface
  - 4 bytes

Array: RL x L x chn

continued on next figure

Figure 603: Data Format Structure for 3DPR, G1, G1
continued from last figure

Figure 604: Data Format Structure for 3DPR, G1

G2_GridHeader
precipRate
rainRate
snowRate
flagHeavyIcePrecip
mixedPhRate
precipRateESurface
precipRateESurface2
precipRateNearSurface
rainRateNearSurface
snowRateNearSurface
mixedPhRateNearSurface
precipWaterIntegrated
precipIceIntegrated
precipRateAve24
2FactorCorrected
2FactorCorrectedESurface
2FactorCorrectedNearSurface
2FactorMeasuredNearSurface
2FactorCorrectedDPR
2FactorCorrectedESurfaceDPR
2FactorCorrectedNearSurfaceDPR
2FactorMeasured
dm

continued on next figure

Figure 605: Data Format Structure for 3DPR, G2, G2
continued from last figure

Figure 606: Data Format Structure for 3DPR, G2

Figure 607: Data Format Structure for 3DPR, G1, precipRate
5.52 3DPR - DPR Full Product

**Figure 608: Data Format Structure for 3DPR, G1, rainRate**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x hgt x rt x st

**Figure 609: Data Format Structure for 3DPR, G1, snowRate**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x hgt x rt x st

**Figure 610: Data Format Structure for 3DPR, G1, flagHeavyIcePrecip**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x rt x st

**Figure 611: Data Format Structure for 3DPR, G1, mixedPhRate**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x hgt x rt x st

**Figure 612: Data Format Structure for 3DPR, G1, precipRateESurface**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x rt x st x bin
Figure 613: Data Format Structure for 3DPR, G1, precipRateESurface2

precipRateESurface2

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

Array: ltL x lnL x chn x rt x st

Figure 614: Data Format Structure for 3DPR, G1, precipRateNearSurface

precipRateNearSurface

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

Array: ltL x lnL x chn x rt x st

Figure 615: Data Format Structure for 3DPR, G1, rainRateNearSurface

rainRateNearSurface

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

Array: ltL x lnL x chn x rt x st
5.52 3DPR - DPR Full Product

snowRateNearSurface
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

mixedPhRateNearSurface
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

precipWaterIntegrated
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

precipIceIntegrated
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

precipRateAve24
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

Figure 616: Data Format Structure for 3DPR, G1, snowRateNearSurface

Figure 617: Data Format Structure for 3DPR, G1, mixedPhRateNearSurface

Figure 618: Data Format Structure for 3DPR, G1, precipWaterIntegrated

Figure 619: Data Format Structure for 3DPR, G1, precipIceIntegrated

Figure 620: Data Format Structure for 3DPR, G1, precipRateAve24
Figure 621: Data Format Structure for 3DPR, G1, zFactorCorrected

zFactorCorrected
- count: 4 bytes, Array: ltL x lnL x inst x hgt x rt x st
- mean: 4 bytes, Array: ltL x lnL x inst x hgt x rt x st
- stdev: 4 bytes, Array: ltL x lnL x inst x hgt x rt x st
- hist: 4 bytes, Array: ltL x lnL x inst x hgt x rt x st x bin

Figure 622: Data Format Structure for 3DPR, G1, zFactorCorrectedESurface

zFactorCorrectedESurface
- count: 4 bytes, Array: ltL x lnL x inst x rt x st
- mean: 4 bytes, Array: ltL x lnL x inst x rt x st
- stdev: 4 bytes, Array: ltL x lnL x inst x rt x st
- hist: 4 bytes, Array: ltL x lnL x inst x rt x st x bin

Figure 623: Data Format Structure for 3DPR, G1, zFactorCorrectedNearSurface

zFactorCorrectedNearSurface
- count: 4 bytes, Array: ltL x lnL x inst x rt x st
- mean: 4 bytes, Array: ltL x lnL x inst x rt x st
- stdev: 4 bytes, Array: ltL x lnL x inst x rt x st
- hist: 4 bytes, Array: ltL x lnL x inst x rt x st x bin

Figure 624: Data Format Structure for 3DPR, G1, zFactorMeasuredNearSurface

zFactorMeasuredNearSurface
- count: 4 bytes, Array: ltL x lnL x inst x rt x st
- mean: 4 bytes, Array: ltL x lnL x inst x rt x st
- stdev: 4 bytes, Array: ltL x lnL x inst x rt x st
- hist: 4 bytes, Array: ltL x lnL x inst x rt x st x bin

Figure 625: Data Format Structure for 3DPR, G1, zFactorCorrectedDPR

zFactorCorrectedDPR
- count: 4 bytes, Array: ltL x lnL x inst x hgt x rt x st
- mean: 4 bytes, Array: ltL x lnL x inst x hgt x rt x st
- stdev: 4 bytes, Array: ltL x lnL x inst x hgt x rt x st
- hist: 4 bytes, Array: ltL x lnL x inst x hgt x rt x st x bin
Figure 626: Data Format Structure for 3DPR, G1, zFactorCorrectedESurfaceDPR

Figure 627: Data Format Structure for 3DPR, G1, zFactorCorrectedNearSurfaceDPR

Figure 628: Data Format Structure for 3DPR, G1, zFactorMeasured

Figure 629: Data Format Structure for 3DPR, G1, dm

Figure 630: Data Format Structure for 3DPR, G1, dBNw
Figure 631: Data Format Structure for 3DPR, G1, epsilonDPR

epsilonDPR
- count: 4 bytes
  - Array: ltL x lnL x inst x hgt x rt x st
- mean: 4 bytes
  - Array: ltL x lnL x inst x hgt x rt x st
- stdev: 4 bytes
  - Array: ltL x lnL x inst x hgt x rt x st
- hist: 4 bytes
  - Array: ltL x lnL x inst x hgt x rt x st x bin

Figure 632: Data Format Structure for 3DPR, G1, epsilon

epsilon
- count: 4 bytes
  - Array: ltL x lnL x inst x rt x st
- mean: 4 bytes
  - Array: ltL x lnL x inst x rt x st
- stdev: 4 bytes
  - Array: ltL x lnL x inst x rt x st
- hist: 4 bytes
  - Array: ltL x lnL x inst x rt x st x bin

Figure 633: Data Format Structure for 3DPR, G1, zeta

zeta
- count: 4 bytes
  - Array: ltL x lnL x inst x ang x rt x st
- mean: 4 bytes
  - Array: ltL x lnL x inst x ang x rt x st
- stdev: 4 bytes
  - Array: ltL x lnL x inst x ang x rt x st
- hist: 4 bytes
  - Array: ltL x lnL x inst x ang x rt x st x bin

Figure 634: Data Format Structure for 3DPR, G1, piaHB

piaHB
- count: 4 bytes
  - Array: ltL x lnL x inst x ang x rt x st
- mean: 4 bytes
  - Array: ltL x lnL x inst x ang x rt x st
- stdev: 4 bytes
  - Array: ltL x lnL x inst x ang x rt x st
- hist: 4 bytes
  - Array: ltL x lnL x inst x ang x rt x st x bin

Figure 635: Data Format Structure for 3DPR, G1, piaHybrid
Figure 636: Data Format Structure for 3DPR, G1, piaHybridDPR

Figure 637: Data Format Structure for 3DPR, G1, piaSRT

Figure 638: Data Format Structure for 3DPR, G1, piaSRTdpr

Figure 639: Data Format Structure for 3DPR, G1, piaFinal

Figure 640: Data Format Structure for 3DPR, G1, piaFinalDPR
5 STANDARD GPM PRODUCTS

Figure 641: Data Format Structure for 3DPR, G1, piaFinalSubset

Figure 642: Data Format Structure for 3DPR, G1, piaFinalDPRsubset

Figure 643: Data Format Structure for 3DPR, G1, heightBB

Figure 644: Data Format Structure for 3DPR, G1, heightBBnadir

Figure 645: Data Format Structure for 3DPR, G1, BBwidthNadir
5.52 3DPR - DPR Full Product

Figure 646: Data Format Structure for 3DPR, G1, heightStormTop

- count 4 bytes Array: ltL x lnL x chn x rt x st
- mean 4 bytes Array: ltL x lnL x chn x rt x st
- stdev 4 bytes Array: ltL x lnL x chn x rt x st
- hist 4 bytes Array: ltL x lnL x chn x rt x st x bin

Figure 647: Data Format Structure for 3DPR, G1, BBwidth

- count 4 bytes Array: ltL x lnL x chn x rt x st
- mean 4 bytes Array: ltL x lnL x chn x rt x st
- stdev 4 bytes Array: ltL x lnL x chn x rt x st
- hist 4 bytes Array: ltL x lnL x chn x rt x st x bin

Figure 648: Data Format Structure for 3DPR, G1, observationCounts

- total 4 bytes Array: ltL x lnL x inst x st
- localTime 4 bytes Array: ltL x lnL x inst x tim x st
- pia 4 bytes Array: ltL x lnL x inst x ang x st
- shallowRain 4 bytes Array: ltL x lnL x inst x st

Figure 649: Data Format Structure for 3DPR, G1, precipRateLocalTime

- count 4 bytes Array: ltL x lnL x chn x tim x st
- mean 4 bytes Array: ltL x lnL x chn x tim x st
- stdev 4 bytes Array: ltL x lnL x chn x tim x st

Figure 650: Data Format Structure for 3DPR, G1, DFRmNearSurface

- count 4 bytes Array: ltL x lnL x rt x st
- mean 4 bytes Array: ltL x lnL x rt x st
- stdev 4 bytes Array: ltL x lnL x rt x st
- hist 4 bytes Array: ltL x lnL x rt x st x bin

Figure 651: Data Format Structure for 3DPR, G1, DFRNearSurface
Figure 652: Data Format Structure for 3DPR, G2, precipRate

Figure 653: Data Format Structure for 3DPR, G2, rainRate

Figure 654: Data Format Structure for 3DPR, G2, snowRate

Figure 655: Data Format Structure for 3DPR, G2, flagHeavyIcePrecip
Figure 656: Data Format Structure for 3DPR, G2, mixedPhRate

Figure 657: Data Format Structure for 3DPR, G2, precipRateESurface

Figure 658: Data Format Structure for 3DPR, G2, precipRateESurface2

Figure 659: Data Format Structure for 3DPR, G2, precipRateNearSurface

Figure 660: Data Format Structure for 3DPR, G2, rainRateNearSurface

Figure 661: Data Format Structure for 3DPR, G2, snowRateNearSurface

Figure 662: Data Format Structure for 3DPR, G2, mixedPhRateNearSurface
5 STANDARD GPM PRODUCTS

Figure 663: Data Format Structure for 3DPR, G2, precipWaterIntegrated

Figure 664: Data Format Structure for 3DPR, G2, precipIceIntegrated

Figure 665: Data Format Structure for 3DPR, G2, precipRateAve24
Figure 666: Data Format Structure for 3DPR, G2, zFactorCorrected

Figure 667: Data Format Structure for 3DPR, G2, zFactorCorrectedESurface

Figure 668: Data Format Structure for 3DPR, G2, zFactorCorrectedNearSurface

Figure 669: Data Format Structure for 3DPR, G2, zFactorMeasuredNearSurface

Figure 670: Data Format Structure for 3DPR, G2, zFactorCorrectedDPR

Figure 671: Data Format Structure for 3DPR, G2, zFactorCorrectedESurfaceDPR

Figure 672: Data Format Structure for 3DPR, G2, zFactorCorrectedNearSurfaceDPR
Figure 673: Data Format Structure for 3DPR, G2, zFactorMeasured

Figure 674: Data Format Structure for 3DPR, G2, dm

Figure 675: Data Format Structure for 3DPR, G2, dBNw
Figure 676: Data Format Structure for 3DPR, G2, epsilonDPR

Figure 677: Data Format Structure for 3DPR, G2, epsilon

Figure 678: Data Format Structure for 3DPR, G2, zeta

Figure 679: Data Format Structure for 3DPR, G2, piaHB

Figure 680: Data Format Structure for 3DPR, G2, piaHybrid

Figure 681: Data Format Structure for 3DPR, G2, piaHybridDPR

Figure 682: Data Format Structure for 3DPR, G2, piaSR T
Figure 683: Data Format Structure for 3DPR, G2, piaSRTdpr

Figure 684: Data Format Structure for 3DPR, G2, piaFinal

Figure 685: Data Format Structure for 3DPR, G2, piaFinalDPR
FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

Grids (Group)
G1 (Grid)

G1_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipRate (Group in G1)
Conditional Precipitation Rate.

count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**hist** (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin): Histogram. Special values are defined as:
-9999 Missing value

**rainRate** (Group in G1)
Conditional liquid water Rain Rate.

**count** (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st): Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x hgt x rt x st): Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x hgt x rt x st): Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin): Histogram. Special values are defined as:
-9999 Missing value

**snowRate** (Group in G1)
Conditional Snowfall Rate.

**count** (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st): Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x hgt x rt x st): Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x hgt x rt x st): Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin): Histogram. Special values are defined as:
-9999 Missing value

**flagHeavyIcePrecip** (Group in G1)
Counts of the occurrence of flagHeavyIcePrecip. Mean and std. dev. are set to missing. The histogram contains counts of the integer flag values, with bins from 1 to 30.
count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

mixedPhRate (Group in G1)
Conditional Mixed Phase Precipitation Rate.

count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipRateESurface (Group in G1)
Conditional Estimated Surface Precipitation Rate.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value
**stddev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**precipRateESurface2** (Group in G1)
Alternate Conditional Estimated Surface Precipitation Rate.

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
Mean. Special values are defined as:
-9999.9  Missing value

**stddev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**precipRateNearSurface** (Group in G1)
Conditional Precipitation Rate at Near Surface Level.

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
Mean. Special values are defined as:
-9999.9  Missing value

**stddev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value
rainRateNearSurface (Group in G1)
Unconditional liquid Rain Rate at Near Surface Level.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

snowRateNearSurface (Group in G1)
Conditional Snow Rate at Near Surface Level.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

mixedPhRateNearSurface (Group in G1)
Conditional Mixed Phase Precipitation Rate at Near Surface Level.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value
mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipWaterIntegrated (Group in G1)
Integrated Precipitable Water (g/m²).

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipIceIntegrated (Group in G1)
Integrated Precipitable Ice

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value
hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipRateAve24 (Group in G1)
Average Precipitation Rate in 24hrs.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

zFactorCorrected (Group in G1)
Corrected Reflectivity

count (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

zFactorCorrectedESurface (Group in G1)
Corrected Reflectivity at the Estimated Surface
count (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
  -9999  Missing value

mean (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
  -9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
  -9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
  -9999  Missing value

\textbf{zFactorCorrectedNearSurface} (Group in G1)
Corrected Reflectivity at the Near Surface Level.

\textbf{count} (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
  -9999  Missing value

\textbf{mean} (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
  -9999.9  Missing value

\textbf{stdev} (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
  -9999.9  Missing value

\textbf{hist} (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
  -9999  Missing value

\textbf{zFactorMeasuredNearSurface} (Group in G1)
Measured Reflectivity at the Near Surface Level.

\textbf{count} (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
  -9999  Missing value

\textbf{mean} (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
  -9999.9  Missing value
stdev (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

zFactorCorrectedDPR (Group in G1)
Corrected Reflectivity from DPR

count (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

zFactorCorrectedESurfaceDPR (Group in G1)
Corrected Reflectivity from DPR at Estimated Surface.

count (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value
**zFactorCorrectedNearSurfaceDPR** (Group in G1)
Corrected Reflectivity from DPR at the Near Surface Level.

**count** (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

---

**zFactorMeasured** (Group in G1)
Measured Reflectivity

**count** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

---

**dm** (Group in G1)

**count** (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value
mean (4-byte float, array size: ltL x lnL x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

dBNw (Group in G1)

count (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

epsilonDPR (Group in G1)

count (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**hist** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**epsilon** (Group in G1)

**count** (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**zeta** (Group in G1)
Integral of 0.2*ln(10)*alpha*Zm^{beta} over the slant range path where alpha and Zm are functions of range.

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**piaHB** (Group in G1)
Hitchfield-Bordan Path Integrated Attenuation for the slant range path.
count (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
    -9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
    -9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
    -9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
    -9999 Missing value

piaHybrid (Group in G1)
Weighted Hybrid PIA between the HB solution and the SRT PIA.

count (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
    -9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
    -9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
    -9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
    -9999 Missing value

piaHybridDPR (Group in G1)
Weighted Hybrid PIA between the HB solution and the SRT PIA for DPR.

count (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
    -9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
    -9999.9 Missing value
**stdev** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\)):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: \(ltL \times lnL \times inst \times ang \times rt \times st \times bin\)):
Histogram. Special values are defined as:
-9999 Missing value

**piaSRT** (Group in G1)
Path Integrated Attenuation from SRT.

**count** (4-byte integer, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\)):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\)):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\)):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: \(ltL \times lnL \times inst \times ang \times rt \times st \times bin\)):
Histogram. Special values are defined as:
-9999 Missing value

**piaSRTdpr** (Group in G1)
Path Integrated Attenuation from SRT DPR

**count** (4-byte integer, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\)):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\)):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\)):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: \(ltL \times lnL \times inst \times ang \times rt \times st \times bin\)):
Histogram. Special values are defined as:
-9999 Missing value
**piaFinal** (Group in G1)
Final Path Integrated Attenuation

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st): Count. Special values are defined as:
- -9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x ang x rt x st): mean. Special values are defined as:
- -9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st): Standard deviation. Special values are defined as:
- -9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin): Histogram. Special values are defined as:
- -9999 Missing value

---

**piaFinalDPR** (Group in G1)
Final Path Integrated Attenuation from DPR

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st): Count. Special values are defined as:
- -9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x ang x rt x st): mean. Special values are defined as:
- -9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st): Standard deviation. Special values are defined as:
- -9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin): Histogram. Special values are defined as:
- -9999 Missing value

---

**piaFinalSubset** (Group in G1)
Final Path Integrated Attenuation Subset

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st): Count. Special values are defined as:
- -9999 Missing value
**mean** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**piaFinalDPRsubset** (Group in G1)
Final Path Integrated Attenuation from DPR Subset

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**heightBB** (Group in G1)
Height of Bright Band.

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**heightBBnadir** (Group in G1)
Height of Bright Band from Nadir.
**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value
**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value
**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**BBwidthNadir** (Group in G1)
Width of Bright Band at Nadir
**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value
**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value
**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**heightStormTop** (Group in G1)
Storm Top Height
count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

BBwidth (Group in G1)
Bright Band Width

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

observationCounts (Group in G1)
Observation Counts

total (4-byte integer, array size: ltL x lnL x inst x st):
Total obs. Special values are defined as:
-9999 Missing value

localTime (4-byte integer, array size: ltL x lnL x inst x tim x st):
obs time. Special values are defined as:
-9999 Missing value
**pia** (4-byte integer, array size: \( l_t L \times l_n L \times i_n t \times a_n g \times s_t \)):
obs PIA. Special values are defined as:
\[-9999 \text{ Missing value}\]

**shallowRain** (4-byte integer, array size: \( l_t L \times l_n L \times i_n t \times s_t \)):
obs time. Special values are defined as:
\[-9999 \text{ Missing value}\]

**precipRateLocalTime** (Group in G1)
Precipitation Rate by Local Time

**count** (4-byte integer, array size: \( l_t L \times l_n L \times c_h n \times t_i m \times s_t \)):
Count. Special values are defined as:
\[-9999 \text{ Missing value}\]

**mean** (4-byte float, array size: \( l_t L \times l_n L \times c_h n \times t_i m \times s_t \)):
mean. Special values are defined as:
\[-9999.9 \text{ Missing value}\]

**stdev** (4-byte float, array size: \( l_t L \times l_n L \times c_h n \times t_i m \times s_t \)):
Standard deviation. Special values are defined as:
\[-9999.9 \text{ Missing value}\]

**DFRmNearSurface** (Group in G1)
DFRm at the Near Surface level

**count** (4-byte integer, array size: \( l_t L \times l_n L \times r_t \times s_t \)):
Count. Special values are defined as:
\[-9999 \text{ Missing value}\]

**mean** (4-byte float, array size: \( l_t L \times l_n L \times r_t \times s_t \)):
mean. Special values are defined as:
\[-9999.9 \text{ Missing value}\]

**stdev** (4-byte float, array size: \( l_t L \times l_n L \times r_t \times s_t \)):
Standard deviation. Special values are defined as:
\[-9999.9 \text{ Missing value}\]

**hist** (4-byte integer, array size: \( l_t L \times l_n L \times r_t \times s_t \times b_i n \)):
Histogram. Special values are defined as:
\[-9999 \text{ Missing value}\]

**DFRNearSurface** (Group in G1)
DFR at the Near Surface level
count (4-byte integer, array size: ltL x lnL x rt x st):
Count. Special values are defined as:
  -9999  Missing value

mean (4-byte float, array size: ltL x lnL x rt x st):
mean. Special values are defined as:
  -9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x rt x st):
Standard deviation. Special values are defined as:
  -9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x rt x st x bin):
Histogram. Special values are defined as:
  -9999  Missing value

precipRateNearSurfaceUnconditional (4-byte float, array size: ltL x lnL x chn):
Rain, not conditioned on rain. Special values are defined as:
  -9999.9  Missing value

precipProbabilityNearSurface (4-byte float, array size: ltL x lnL x chn):
Probability of rain. Special values are defined as:
  -9999.9  Missing value

G2 (Grid)

G2_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipRate (Group in G2)
Conditional Precipitation Rate

count (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
  -9999  Missing value

mean (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
  -9999.9  Missing value

stdev (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
  -9999.9  Missing value
rainRate (Group in G2)
Conditional Liquid Rain Rate

count (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

snowRate (Group in G2)
Conditional Snow Rate

count (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

flagHeavyIcePrecip (Group in G2)
Counts of the occurrence of flagHeavyIcePrecip. Mean and std. dev. are set to missing.
The histogram contains counts of the integer flag values, with bins from 1 to 30.

count (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
mixedPhRate (Group in G2)
Conditional Precipitation Rate of Mixed Phase

*count* (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

*mean* (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

*stdev* (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

precipRateESurface (Group in G2)
Conditional Estimated Precipitation Rate at the Surface

*count* (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

*mean* (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

*stdev* (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

precipRateESurface2 (Group in G2)
Alternate Conditional Estimated Precipitation Rate at the Surface

*count* (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

*mean* (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

*stdev* (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**precipRateNearSurface** (Group in G2)
Conditional Precipitation Rate at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**rainRateNearSurface** (Group in G2)
Conditional Liquid Rain Rate at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**snowRateNearSurface** (Group in G2)
Conditional Snow Rate at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**mixedPhRateNearSurface** (Group in G2)
Conditional Precipitation Rate of Mixed Phase at the Near Surface Level.

- **count** (4-byte integer, array size: ltH x lnH x chn x rt):
  Count. Special values are defined as:
  -9999  Missing value

- **mean** (4-byte float, array size: ltH x lnH x chn x rt):
  mean. Special values are defined as:
  -9999.9  Missing value

- **stdev** (4-byte float, array size: ltH x lnH x chn x rt):
  Standard deviation. Special values are defined as:
  -9999.9  Missing value

**precipWaterIntegrated** (Group in G2)
Integrated Precipitable Water ($g/m^2$).

- **count** (4-byte integer, array size: ltH x lnH x chn x rt):
  Count. Special values are defined as:
  -9999  Missing value

- **mean** (4-byte float, array size: ltH x lnH x chn x rt):
  mean. Special values are defined as:
  -9999.9  Missing value

- **stdev** (4-byte float, array size: ltH x lnH x chn x rt):
  Standard deviation. Special values are defined as:
  -9999.9  Missing value

**precipIceIntegrated** (Group in G2)
Integrated Precipitable Ice

- **count** (4-byte integer, array size: ltH x lnH x chn x rt):
  Count. Special values are defined as:
  -9999  Missing value

- **mean** (4-byte float, array size: ltH x lnH x chn x rt):
  mean. Special values are defined as:
  -9999.9  Missing value

- **stdev** (4-byte float, array size: ltH x lnH x chn x rt):
  Standard deviation. Special values are defined as:
  -9999.9  Missing value
precipRateAve24 (Group in G2)
Conditional Precipitation Rate Averaged for 24hrs.

count (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
  -9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
  -9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
  -9999.9 Missing value

zFactorCorrected (Group in G2)
Corrected Reflectivity.

count (4-byte integer, array size: ltH x lnH x inst x hgt x rt):
Count. Special values are defined as:
  -9999 Missing value

mean (4-byte float, array size: ltH x lnH x inst x hgt x rt):
mean. Special values are defined as:
  -9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x inst x hgt x rt):
Standard deviation. Special values are defined as:
  -9999.9 Missing value

zFactorCorrectedESurface (Group in G2)
Corrected Reflectivity Estimate at the Surface

count (4-byte integer, array size: ltH x lnH x inst x rt):
Count. Special values are defined as:
  -9999 Missing value

mean (4-byte float, array size: ltH x lnH x inst x rt):
mean. Special values are defined as:
  -9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x inst x rt):
Standard deviation. Special values are defined as:
  -9999.9 Missing value
**zFactorCorrectedNearSurface** (Group in G2)
Corrected Reflectivity at the Near Surface Level.

- **count** (4-byte integer, array size: ltH x lnH x inst x rt):
  Count. Special values are defined as:
  -9999 Missing value

- **mean** (4-byte float, array size: ltH x lnH x inst x rt):
  mean. Special values are defined as:
  -9999.9 Missing value

- **stdev** (4-byte float, array size: ltH x lnH x inst x rt):
  Standard deviation. Special values are defined as:
  -9999.9 Missing value

**zFactorMeasuredNearSurface** (Group in G2)
Measured Reflectivity at the Near Surface Level.

- **count** (4-byte integer, array size: ltH x lnH x inst x rt):
  Count. Special values are defined as:
  -9999 Missing value

- **mean** (4-byte float, array size: ltH x lnH x inst x rt):
  mean. Special values are defined as:
  -9999.9 Missing value

- **stdev** (4-byte float, array size: ltH x lnH x inst x rt):
  Standard deviation. Special values are defined as:
  -9999.9 Missing value

**zFactorCorrectedDPR** (Group in G2)
Corrected Reflectivity from DPR

- **count** (4-byte integer, array size: ltH x lnH x inst x hgt x rt):
  Count. Special values are defined as:
  -9999 Missing value

- **mean** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
  mean. Special values are defined as:
  -9999.9 Missing value

- **stdev** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
  Standard deviation. Special values are defined as:
  -9999.9 Missing value
**zFactorCorrectedESurfaceDPR** (Group in G2)
Estimated Corrected Reflectivity at the Surface

**count** (4-byte integer, array size: ltH x lnH x inst x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurfaceDPR** (Group in G2)
Corrected Reflectivity at the Near Surface Level for DPR

**count** (4-byte integer, array size: ltH x lnH x inst x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**zFactorMeasured** (Group in G2)
Corrected Reflectivity

**count** (4-byte integer, array size: ltH x lnH x inst x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
dm (Group in G2)
Mean Mass-Weighted Drop Diameter

count (4-byte integer, array size: ltH x lnH x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value


dBNw (Group in G2)
Normalized Drop Concentration Parameter

count (4-byte integer, array size: ltH x lnH x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value


epsilonDPR (Group in G2)

count (4-byte integer, array size: ltH x lnH x inst x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x inst x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x inst x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**epsilon (Group in G2)**

- **count** (4-byte integer, array size: ltH x lnH x inst x rt): Count. Special values are defined as:
  - -9999 Missing value
- **mean** (4-byte float, array size: ltH x lnH x inst x rt): mean. Special values are defined as:
  - -9999.9 Missing value
- **stdev** (4-byte float, array size: ltH x lnH x inst x rt): Standard deviation. Special values are defined as:
  - -9999.9 Missing value

**zeta (Group in G2)**

Integral of $0.2*\ln(10)*\alpha*Z_{m}^{beta}$ over the slant range path where $\alpha$ and $Z_{m}$ are functions of range.

- **count** (4-byte integer, array size: ltH x lnH x inst x ang x rt): Count. Special values are defined as:
  - -9999 Missing value
- **mean** (4-byte float, array size: ltH x lnH x inst x ang x rt): mean. Special values are defined as:
  - -9999.9 Missing value
- **stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt): Standard deviation. Special values are defined as:
  - -9999.9 Missing value

**piaHB (Group in G2)**

Hitchfield-Bordan Path Integrated Attenuation for the slant range path.

- **count** (4-byte integer, array size: ltH x lnH x inst x ang x rt): Count. Special values are defined as:
  - -9999 Missing value
- **mean** (4-byte float, array size: ltH x lnH x inst x ang x rt): mean. Special values are defined as:
  - -9999.9 Missing value
- **stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt): Standard deviation. Special values are defined as:
  - -9999.9 Missing value
**piaHybrid** (Group in G2)
Weighted Hybrid PIA between the HB solution and the SRT PIA.

- **count** (4-byte integer, array size: ltH x lnH x inst x ang x rt):
  Count. Special values are defined as:
  -9999 Missing value

- **mean** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  mean. Special values are defined as:
  -9999.9 Missing value

- **stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  Standard deviation. Special values are defined as:
  -9999.9 Missing value

**piaHybridDPR** (Group in G2)
Weighted Hybrid PIA between the HB solution and the SRT PIA for DPR.

- **count** (4-byte integer, array size: ltH x lnH x inst x ang x rt):
  Count. Special values are defined as:
  -9999 Missing value

- **mean** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  mean. Special values are defined as:
  -9999.9 Missing value

- **stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  Standard deviation. Special values are defined as:
  -9999.9 Missing value

**piaSRT** (Group in G2)
Path Integrated Attenuation from SRT.

- **count** (4-byte integer, array size: ltH x lnH x inst x ang x rt):
  Count. Special values are defined as:
  -9999 Missing value

- **mean** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  mean. Special values are defined as:
  -9999.9 Missing value

- **stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  Standard deviation. Special values are defined as:
  -9999.9 Missing value
**piaSRTdpr** (Group in G2)
Path Integrated Attenuation from SRT for DPR.

**count** (4-byte integer, array size: $ltH \times lnH \times inst \times ang \times rt$):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: $ltH \times lnH \times inst \times ang \times rt$):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: $ltH \times lnH \times inst \times ang \times rt$):
Standard deviation. Special values are defined as:
-9999.9 Missing value

---

**piaFinal** (Group in G2)
Final Path Integrated Attenuation Estimate.

**count** (4-byte integer, array size: $ltH \times lnH \times inst \times ang \times rt$):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: $ltH \times lnH \times inst \times ang \times rt$):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: $ltH \times lnH \times inst \times ang \times rt$):
Standard deviation. Special values are defined as:
-9999.9 Missing value

---

**piaFinalDPR** (Group in G2)
Final Path Integrated Attenuation Estimate for DPR.

**count** (4-byte integer, array size: $ltH \times lnH \times inst \times ang \times rt$):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: $ltH \times lnH \times inst \times ang \times rt$):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: $ltH \times lnH \times inst \times ang \times rt$):
Standard deviation. Special values are defined as:
-9999.9 Missing value
heightBB (Group in G2)
Height of the Bright Band.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

heightStormTop (Group in G2)
Height of the Storm Top.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

BBwidth (Group in G2)
Bright Band Width.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**observationCounts** (Group in G2)
Observation Counts.

**total** (4-byte integer, array size: \(ltH \times lnH \times inst\)):
Total obs. Special values are defined as:
- `-9999`  Missing value

**pia** (4-byte integer, array size: \(ltH \times lnH \times inst \times ang\)):
obs PIA. Special values are defined as:
- `-9999`  Missing value

**shallowRain** (4-byte integer, array size: \(ltH \times lnH \times inst\)):
obs time. Special values are defined as:
- `-9999`  Missing value

**DFRmNearSurface** (Group in G2)
DFRm at the Near Surface level

**count** (4-byte integer, array size: \(ltH \times lnH \times rt\)):
Count. Special values are defined as:
- `-9999`  Missing value

**mean** (4-byte float, array size: \(ltH \times lnH \times rt\)):
mean. Special values are defined as:
- `-9999.9`  Missing value

**stdev** (4-byte float, array size: \(ltH \times lnH \times rt\)):
Standard deviation. Special values are defined as:
- `-9999.9`  Missing value

**DFRNearSurface** (Group in G2)
DFR at the Near Surface level

**count** (4-byte integer, array size: \(ltH \times lnH \times rt\)):
Count. Special values are defined as:
- `-9999`  Missing value

**mean** (4-byte float, array size: \(ltH \times lnH \times rt\)):
mean. Special values are defined as:
- `-9999.9`  Missing value

**stdev** (4-byte float, array size: \(ltH \times lnH \times rt\)):
Standard deviation. Special values are defined as:
- `-9999.9`  Missing value
precipRateNearSurfaceUnconditional (4-byte float, array size: ltH x lnH x chn): Rain, not conditioned on rain. Special values are defined as:
  -9999.9 Missing value

precipProbabilityNearSurface (4-byte float, array size: ltH x lnH x chn): Probability of rain. Special values are defined as:
  -9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_3DPR_H_
define _TK_3DPR_H_

#ifndef _L3DPR_G2_DFRNEARSURFACE_
define _L3DPR_G2_DFRNEARSURFACE_

typedef struct {
    int count[3][1440][536];
    float mean[3][1440][536];
    float stdev[3][1440][536];
} L3DPR_G2_DFRNEARSURFACE;
#endif

#ifndef _L3DPR_G2_DFRMNEARSURFACE_
define _L3DPR_G2_DFRMNEARSURFACE_

typedef struct {
    int count[3][1440][536];
    float mean[3][1440][536];
    float stdev[3][1440][536];
} L3DPR_G2_DFRMNEARSURFACE;
#endif

#ifndef _L3DPR_G2_OBSERVATIONCOUNTS_
define _L3DPR_G2_OBSERVATIONCOUNTS_

typedef struct {
    int total[4][1440][536];
    int pia[7][4][1440][536];
    int shallowRain[4][1440][536];
} L3DPR_G2_OBSERVATIONCOUNTS;
```

#endif

#ifndef _L3DPR_G2_BBWIDTH_
define _L3DPR_G2_BBWIDTH_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_BBWIDTH;
#endif

#ifndef _L3DPR_G2_HEIGHTSTORMTOP_
define _L3DPR_G2_HEIGHTSTORMTOP_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_HEIGHTSTORMTOP;
#endif

#ifndef _L3DPR_G2_HEIGHTBB_
define _L3DPR_G2_HEIGHTBB_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_HEIGHTBB;
#endif

#ifndef _L3DPR_G2_PIAFINALDPR_
define _L3DPR_G2_PIAFINALDPR_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPR_G2_PIAFINALDPR;
#endif
#ifndef _L3DPR_G2_PIAFINAL_
#define _L3DPR_G2_PIAFINAL_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPR_G2_PIAFINAL;

#endif

#ifndef _L3DPR_G2_PIASRTDPR_
#define _L3DPR_G2_PIASRTDPR_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPR_G2_PIASRTDPR;

#endif

#ifndef _L3DPR_G2_PIAHYBRIDDPR_
#define _L3DPR_G2_PIAHYBRIDDPR_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPR_G2_PIAHYBRIDDPR;

#endif
typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPR_G2_PIAHYBRID;

#endif

#ifndef _L3DPR_G2_PIAHB_
#define _L3DPR_G2_PIAHB_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPR_G2_PIAHB;

#endif

#ifndef _L3DPR_G2_ZETA_
#define _L3DPR_G2_ZETA_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPR_G2_ZETA;

#endif

#ifndef _L3DPR_G2_EPSILON_
#define _L3DPR_G2_EPSILON_

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
} L3DPR_G2_EPSILON;
typedef struct {
    int count[3][5][4][1440][536];
    float mean[3][5][4][1440][536];
    float stdev[3][5][4][1440][536];
} L3DPR_G2_EPSILON;
#endif

#define _L3DPR_G2_EPSILON DPR_

typedef struct {
    int count[3][5][4][1440][536];
    float mean[3][5][4][1440][536];
    float stdev[3][5][4][1440][536];
} L3DPR_G2_EPSILON DPR;
#endif

#define _L3DPR_G2_DBNW_ 

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_DBNW;
#endif

#define _L3DPR_G2_DM_ 

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_DM;
#endif

#define _L3DPR_G2_ZFACTORMEASURED_ 

typedef struct {
    int count[3][5][4][1440][536];
typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3DPR_G2_ZFACTORCORRECTEDNEARSURFACEDPR;

#endif

#ifndef _L3DPR_G2_ZFACTORCORRECTEDDESSURFACEDPR_
#define _L3DPR_G2_ZFACTORCORRECTEDDESSURFACEDPR_

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3DPR_G2_ZFACTORCORRECTEDDESSURFACEDPR;

#endif

#ifndef _L3DPR_G2_ZFACTORCORRECTEDDPR_
#define _L3DPR_G2_ZFACTORCORRECTEDDPR_

typedef struct {
    int count[3][5][4][1440][536];
    float mean[3][5][4][1440][536];
    float stdev[3][5][4][1440][536];
} L3DPR_G2_ZFACTORCORRECTEDDPR;

#endif

#ifndef _L3DPR_G2_ZFACTORMEASUREDNMEARSURFACE_
#define _L3DPR_G2_ZFACTORMEASUREDNMEARSURFACE_

typedef struct {

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3DPR_G2_ZFACTORMEASUREDNEARSURFACE;

#endif

#ifndef _L3DPR_G2_ZFACTORCORRECTEDNEARSURFACE_
define _L3DPR_G2_ZFACTORCORRECTEDNEARSURFACE_

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3DPR_G2_ZFACTORCORRECTEDNEARSURFACE;
#endif

#ifndef _L3DPR_G2_ZFACTORCORRECTEDESURFACE_
define _L3DPR_G2_ZFACTORCORRECTEDESURFACE_

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3DPR_G2_ZFACTORCORRECTEDESURFACE;
#endif

#ifndef _L3DPR_G2_ZFACTORCORRECTED_
define _L3DPR_G2_ZFACTORCORRECTED_

typedef struct {
    int count[3][5][4][1440][536];
    float mean[3][5][4][1440][536];
    float stdev[3][5][4][1440][536];
} L3DPR_G2_ZFACTORCORRECTED;
#endif

#ifndef _L3DPR_G2_PRECIPRATEAVE24_
define _L3DPR_G2_PRECIPRATEAVE24_

typedef struct {
  int count[3][5][1440][536];
  float mean[3][5][1440][536];
  float stdev[3][5][1440][536];
} L3DPR_G2_PRECIPRATEAVE24;

#endif

ifndef _L3DPR_G2_PRECIPICEINTEGRATED_
define _L3DPR_G2_PRECIPICEINTEGRATED_

typedef struct {
  int count[3][5][1440][536];
  float mean[3][5][1440][536];
  float stdev[3][5][1440][536];
} L3DPR_G2_PRECIPICEINTEGRATED;

#endif

ifndef _L3DPR_G2_PRECIPWATERINTEGRATED_
define _L3DPR_G2_PRECIPWATERINTEGRATED_

typedef struct {
  int count[3][5][1440][536];
  float mean[3][5][1440][536];
  float stdev[3][5][1440][536];
} L3DPR_G2_PRECIPWATERINTEGRATED;

#endif

ifndef _L3DPR_G2_MIXEDPHRATENEARSURFACE_
define _L3DPR_G2_MIXEDPHRATENEARSURFACE_

typedef struct {
  int count[3][5][1440][536];
  float mean[3][5][1440][536];
  float stdev[3][5][1440][536];
} L3DPR_G2_MIXEDPHRATENEARSURFACE;

#endif

ifndef _L3DPR_G2_SNOWRATENEARSURFACE_
define _L3DPR_G2_SNOWRATENEARSURFACE_

typedef struct {
  int count[3][5][1440][536];
  float mean[3][5][1440][536];
  float stdev[3][5][1440][536];
} L3DPR_G2_SNOWRATENEARSURFACE;
typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_SNOWRATENEARSURFACE;

#endif

#ifndef _L3DPR_G2_RAINRATENEARSURFACE_
#define _L3DPR_G2_RAINRATENEARSURFACE_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_RAINRATENEARSURFACE;

#endif

#ifndef _L3DPR_G2_PRECIPRATENEARSURFACE_
#define _L3DPR_G2_PRECIPRATENEARSURFACE_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_PRECIPRATENEARSURFACE;

#endif

#ifndef _L3DPR_G2_PRECIPRATEESURFACE2_
#define _L3DPR_G2_PRECIPRATEESURFACE2_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_PRECIPRATEESURFACE2;

#endif

#ifndef _L3DPR_G2_PRECIPRATEESURFACE_
#define _L3DPR_G2_PRECIPRATEESURFACE_

#define _L3DPR_G2_PRECIPRATESURFACE_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_PRECIPRATESURFACE;

#endif

#ifndef _L3DPR_G2_MIXEDPHRATE_
#define _L3DPR_G2_MIXEDPHRATE_

typedef struct {
    int count[3][5][5][1440][536];
    float mean[3][5][5][1440][536];
    float stdev[3][5][5][1440][536];
} L3DPR_G2_MIXEDPHRATE;

#endif

#ifndef _L3DPR_G2_FLAGHEAVYICEPRECIP_
#define _L3DPR_G2_FLAGHEAVYICEPRECIP_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3DPR_G2_FLAGHEAVYICEPRECIP;

#endif

#ifndef _L3DPR_G2_SNOWRATE_
#define _L3DPR_G2_SNOWRATE_

typedef struct {
    int count[3][5][5][1440][536];
    float mean[3][5][5][1440][536];
    float stdev[3][5][5][1440][536];
} L3DPR_G2_SNOWRATE;

#endif
```c
#ifndef _L3DPR_G2_RAINRATE_
define _L3DPR_G2_RAINRATE_

typedef struct {
    int count[3][5][5][1440][536];
    float mean[3][5][5][1440][536];
    float stdev[3][5][5][1440][536];
} L3DPR_G2_RAINRATE;
#endif

#ifndef _L3DPR_G2_PRECIPRATE_
define _L3DPR_G2_PRECIPRATE_

typedef struct {
    int count[3][5][5][1440][536];
    float mean[3][5][5][1440][536];
    float stdev[3][5][5][1440][536];
} L3DPR_G2_PRECIPRATE;
#endif

#ifndef _L3DPR_G2_
define _L3DPR_G2_

typedef struct {
    L3DPR_G2_PRECIPRATE precipRate;
    L3DPR_G2_RAINRATE rainRate;
    L3DPR_G2_SNOWRATE snowRate;
    L3DPR_G2_FLAGHEAVYICEPRECIP flagHeavyIcePrecip;
    L3DPR_G2_MIXEDPHRATE mixedPhRate;
    L3DPR_G2_PRECIPRATEESURFACE precipRateESurface;
    L3DPR_G2_PRECIPRATEESURFACE2 precipRateESurface2;
    L3DPR_G2_PRECIPRATENEARSURFACE precipRateNearSurface;
    L3DPR_G2_RAINRATENEARSURFACE rainRateNearSurface;
    L3DPR_G2_SNOWRATENEARSURFACE snowRateNearSurface;
    L3DPR_G2_MIXEDPHRATENEARSURFACE mixedPhRateNearSurface;
    L3DPR_G2_PRECIPWATERINTEGRATED precipWaterIntegrated;
    L3DPR_G2_PRECIPICEINTEGRATED precipIceIntegrated;
    L3DPR_G2_PRECIPRateAve24 precipRateAve24;
    L3DPR_G2_ZFACTORCORRECTED zFactorCorrected;
    L3DPR_G2_ZFACTORCORRECTEDESURFACE zFactorCorrectedESurface;
    L3DPR_G2_ZFACTORCORRECTEDNEARSURFACE zFactorCorrectedNearSurface;
}
```

5.52 3DPR - DPR Full Product
L3DPR_G2_ZFACTORMEASUREDNEARSURFACE zFactorMeasuredNearSurface;
L3DPR_G2_ZFACTORCORRECTEDDPR zFactorCorrectedDPR;
L3DPR_G2_ZFACTORCORRECTEDESURFACEDPR zFactorCorrectedESurfaceDPR;
L3DPR_G2_ZFACTORCORRECTEDNEARSURFACEDPR zFactorCorrectedNearSurfaceDPR;
L3DPR_G2_ZFACTORMEASURED zFactorMeasured;
L3DPR_G2_DM dm;
L3DPR_G2_DBNW dBNw;
L3DPR_G2_EPSILONDPR epsilonDPR;
L3DPR_G2_EPSILON epsilon;
L3DPR_G2_ZETA zeta;
L3DPR_G2_PIAHB piaHB;
L3DPR_G2_PIAHYBRID piaHybrid;
L3DPR_G2_PIAHYBRIDDPR piaHybridDPR;
L3DPR_G2_PIASRT piaSRT;
L3DPR_G2_PIASRTDPR piaSRTdpr;
L3DPR_G2_PIAFINAL piaFinal;
L3DPR_G2_PIAFINALDPR piaFinalDPR;
L3DPR_G2_HEIGHTBB heightBB;
L3DPR_G2_HEIGHTSTORMTOP heightStormTop;
L3DPR_G2_BBWIDTH BBwidth;
L3DPR_G2_OBSERVATIONCOUNTS observationCounts;
L3DPR_G2_DFRMNEARSURFACE DFRmNearSurface;
L3DPR_G2_DFRNEARSURFACE DFRNearSurface;
float precipRateNearSurfaceUnconditional[5][1440][536];
float precipProbabilityNearSurface[5][1440][536];
} L3DPR_G2;

#endif

#ifndef _L3DPR_G1_DFRNEARSURFACE_
#define _L3DPR_G1_DFRNEARSURFACE_

typedef struct {
    int count[3][3][72][28];
    float mean[3][3][72][28];
    float stdev[3][3][72][28];
    int hist[30][3][3][72][28];
} L3DPR_G1_DFRNEARSURFACE;

#endif

#ifndef _L3DPR_G1_DFRMNEARSURFACE_
#define _L3DPR_G1_DFRMNEARSURFACE_

#endif
typedef struct {
    int count[3][3][72][28];
    float mean[3][3][72][28];
    float stdev[3][3][72][28];
    int hist[30][3][3][72][28];
} L3DPR_G1_DFRMNEARSURFACE;

#endif

#ifndef _L3DPR_G1_PRECIPRATELOCALTIME_
#define _L3DPR_G1_PRECIPRATELOCALTIME_

typedef struct {
    int count[3][24][5][72][28];
    float mean[3][24][5][72][28];
    float stdev[3][24][5][72][28];
} L3DPR_G1_PRECIPRATELOCALTIME;
#endif

#ifndef _L3DPR_G1_OBSERVATIONCOUNTS_
#define _L3DPR_G1_OBSERVATIONCOUNTS_

typedef struct {
    int total[3][4][72][28];
    int localTime[3][24][4][72][28];
    int pia[3][7][4][72][28];
    int shallowRain[3][4][72][28];
} L3DPR_G1_OBSERVATIONCOUNTS;
#endif

#ifndef _L3DPR_G1_BBWIDTH_
#define _L3DPR_G1_BBWIDTH_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_BBWIDTH;
#ifndef _L3DPR_G1_HEIGHTSTORMTOP_
#define _L3DPR_G1_HEIGHTSTORMTOP_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_HEIGHTSTORMTOP;
#endif

#ifndef _L3DPR_G1_BBWIDTHNADIR_
#define _L3DPR_G1_BBWIDTHNADIR_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_BBWIDTHNADIR;
#endif

#ifndef _L3DPR_G1_HEIGHTBBNADIR_
#define _L3DPR_G1_HEIGHTBBNADIR_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_HEIGHTBBNADIR;
#endif

#ifndef _L3DPR_G1_HEIGHTBB_
#define _L3DPR_G1_HEIGHTBB_

typedef struct {
    int count[3][3][5][72][28];
} L3DPR_G1_HEIGHTBB;
float mean[3][3][5][72][28];
float stdev[3][3][5][72][28];
int hist[30][3][3][5][72][28];
} L3DPR_G1_HEIGHTBB;

#endif

#ifndef _L3DPR_G1_PIAFINALDPRSUBSET_
#define _L3DPR_G1_PIAFINALDPRSUBSET_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1_PIAFINALDPRSUBSET;
#endif

#ifndef _L3DPR_G1_PIAFINALSUBSET_
#define _L3DPR_G1_PIAFINALSUBSET_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1_PIAFINALSUBSET;
#endif

#ifndef _L3DPR_G1_PIAFINALDPR_
#define _L3DPR_G1_PIAFINALDPR_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1_PIAFINALDPR;
#endif
#ifndef _L3DPR_G1_PIAFINAL_
#define _L3DPR_G1_PIAFINAL_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1_PIAFINAL;
#endif

#ifndef _L3DPR_G1.PIASRTDPR_
#define _L3DPR_G1.PIASRTDPR_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1.PIASRTDPR;
#endif

#ifndef _L3DPR_G1.PIASRT_
#define _L3DPR_G1.PIASRT_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1.PIASRT;
#endif

#ifndef _L3DPR_G1.PIAHYBRIDDPR_
#define _L3DPR_G1.PIAHYBRIDDPR_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1.PIAHYBRIDDPR;
#endif
typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1_PIAHYBRID;

#endif

#ifndef _L3DPR_G1_PIAHB_
#define _L3DPR_G1_PIAHB_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1_PIAHB;

#endif

#ifndef _L3DPR_G1_ZETA_
#define _L3DPR_G1_ZETA_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPR_G1_ZETA;

#endif

#ifndef _L3DPR_G1_EPSILON_
#define _L3DPR_G1_EPSILON_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3DPR_G1_EPSILON;

#endif

#ifndef _L3DPR_G1_EPSILON_DPR_
#define _L3DPR_G1_EPSILON_DPR_

typedef struct {
    int count[3][3][5][4][72][28];
    float mean[3][3][5][4][72][28];
    float stdev[3][3][5][4][72][28];
    int hist[30][3][3][5][4][72][28];
} L3DPR_G1_EPSILON_DPR;

#endif

#ifndef _L3DPR_G1_DBNW_
#define _L3DPR_G1_DBNW_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_DBNW;

#endif

#ifndef _L3DPR_G1_DM_
#define _L3DPR_G1_DM_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_DM;
#endif

#ifndef _L3DPR_G1_ZFACTORMEASURED_
define _L3DPR_G1_ZFACTORMEASURED_

typedef struct {
    int count[3][3][5][4][72][28];
    float mean[3][3][5][4][72][28];
    float stdev[3][3][5][4][72][28];
    int hist[30][3][3][5][4][72][28];
} L3DPR_G1_ZFACTORMEASURED;
#endif

#ifndef _L3DPR_G1_ZFACTORCORRECTEDNEARSURFACEDPR_
define _L3DPR_G1_ZFACTORCORRECTEDNEARSURFACEDPR_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3DPR_G1_ZFACTORCORRECTEDNEARSURFACEDPR;
#endif

#ifndef _L3DPR_G1_ZFACTORCORRECTEDSURFACEDPR_
define _L3DPR_G1_ZFACTORCORRECTEDSURFACEDPR_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3DPR_G1_ZFACTORCORRECTEDSURFACEDPR;
#endif

#ifndef _L3DPR_G1_ZFACTORCORRECTEDDPR_
define _L3DPR_G1_ZFACTORCORRECTEDDPR_

typedef struct {

}
int count[3][3][5][4][72][28];
float mean[3][3][5][4][72][28];
float stdev[3][3][5][4][72][28];
int hist[30][3][3][5][4][72][28];
} L3DPR_G1_ZFACTORCORRECTEDDPR;

#endif

#ifndef _L3DPR_G1_ZFACTORMEASUREDNEARSURFACE_
#define _L3DPR_G1_ZFACTORMEASUREDNEARSURFACE_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3DPR_G1_ZFACTORMEASUREDNEARSURFACE;

#endif

#ifndef _L3DPR_G1_ZFACTORCORRECTEDNEARSURFACE_
#define _L3DPR_G1_ZFACTORCORRECTEDNEARSURFACE_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3DPR_G1_ZFACTORCORRECTEDNEARSURFACE;

#endif

#ifndef _L3DPR_G1_ZFACTORCORRECTEDSURFACE_
#define _L3DPR_G1_ZFACTORCORRECTEDSURFACE_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3DPR_G1_ZFACTORCORRECTEDSURFACE;

#endif
```c
#ifndef _L3DPR_G1_ZFACTORCORRECTED_
define _L3DPR_G1_ZFACTORCORRECTED_

typedef struct {
    int count[3][3][5][4][72][28];
    float mean[3][3][5][4][72][28];
    float stdev[3][3][5][4][72][28];
    int hist[30][3][3][5][4][72][28];
} L3DPR_G1_ZFACTORCORRECTED;
#endif

#ifndef _L3DPR_G1_PRECIPRATEAVE24_
define _L3DPR_G1_PRECIPRATEAVE24_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_PRECIPRATEAVE24;
#endif

#ifndef _L3DPR_G1_PRECIPICEINTEGRATED_
define _L3DPR_G1_PRECIPICEINTEGRATED_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_PRECIPICEINTEGRATED;
#endif

#ifndef _L3DPR_G1_PRECIPWATERINTEGRATED_
define _L3DPR_G1_PRECIPWATERINTEGRATED_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
```
typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_MIXEDPHRATENEARSURFACE;

#endif

#ifndef _L3DPR_G1_MIXEDPHRATENEARSURFACE_
#define _L3DPR_G1_MIXEDPHRATENEARSURFACE_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_MIXEDPHRATENEARSURFACE;

#endif

#ifndef _L3DPR_G1_SNOWRATENEARSURFACE_
#define _L3DPR_G1_SNOWRATENEARSURFACE_

typedef struct {
    int count[3][3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_SNOWRATENEARSURFACE;

#endif

#ifndef _L3DPR_G1_RAINRATENEARSURFACE_
#define _L3DPR_G1_RAINRATENEARSURFACE_

typedef struct {
    int count[3][3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_RAINRATENEARSURFACE;

#endif

#ifndef _L3DPR_G1_PRECIPRATENEARSURFACE_
#define _L3DPR_G1_PRECIPRATENEARSURFACE_

float stdev[3][3][3][5][72][28];
int hist[30][3][3][5][72][28];

} L3DPR_G1_PRECIPWATERINTEGRATED;

#endif
#define _L3DPR_G1_PRECIPRATENEARSURFACE_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_PRECIPRATENEARSURFACE;

#endif

#ifndef _L3DPR_G1_PRECIPRATEESURFACE2_
#define _L3DPR_G1_PRECIPRATEESURFACE2_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_PRECIPRATEESURFACE2;

#endif

#ifndef _L3DPR_G1_PRECIPRATEESURFACE_
#define _L3DPR_G1_PRECIPRATEESURFACE_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_PRECIPRATEESURFACE;

#endif

#ifndef _L3DPR_G1_MIXEDPHRATE_
#define _L3DPR_G1_MIXEDPHRATE_

typedef struct {
    int count[3][3][5][5][72][28];
    float mean[3][3][5][5][72][28];
    float stdev[3][3][5][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_MIXEDPHRATE;

#endif
} L3DPR_G1_MIXEDPHRATE;

#endif

#ifndef _L3DPR_G1_FLAGHEAVYICEPRECIP_
define _L3DPR_G1_FLAGHEAVYICEPRECIP_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3DPR_G1_FLAGHEAVYICEPRECIP;
#endif

#ifndef _L3DPR_G1_SNOWRATE_
define _L3DPR_G1_SNOWRATE_

typedef struct {
    int count[3][3][5][5][72][28];
    float mean[3][3][5][5][72][28];
    float stdev[3][3][5][5][72][28];
    int hist[30][3][3][5][5][72][28];
} L3DPR_G1_SNOWRATE;
#endif

#ifndef _L3DPR_G1_RAINRATE_
define _L3DPR_G1_RAINRATE_

typedef struct {
    int count[3][3][5][5][72][28];
    float mean[3][3][5][5][72][28];
    float stdev[3][3][5][5][72][28];
    int hist[30][3][3][5][5][72][28];
} L3DPR_G1_RAINRATE;
#endif

#ifndef _L3DPR_G1_PRECIPRATE_
define _L3DPR_G1_PRECIPRATE_

typedef struct {
    int count[3][3][5][5][72][28];
    float mean[3][3][5][5][72][28];
    float stdev[3][3][5][5][72][28];
    int hist[30][3][3][5][5][72][28];
} L3DPR_G1_PRECIPRATE;

#endif

#ifndef _L3DPR_G1_
#define _L3DPR_G1_

typedef struct {
    L3DPR_G1_PRECIPRATE precipRate;
    L3DPR_G1_RAINRATE rainRate;
    L3DPR_G1_SNOWRATE snowRate;
    L3DPR_G1_FLAGHEAVYICEPRECIP flagHeavyIcePrecip;
    L3DPR_G1_MIXEDPHRATE mixedPhRate;
    L3DPR_G1_PRECIPRATEESURFACE precipRateESurface;
    L3DPR_G1_PRECIPRATEESURFACE2 precipRateESurface2;
    L3DPR_G1_PRECIPRATENEARSURFACE precipRateNearSurface;
    L3DPR_G1_RAINRATENEARSURFACE rainRateNearSurface;
    L3DPR_G1_SNOWRATENEARSURFACE snowRateNearSurface;
    L3DPR_G1_MIXEDPHRATENEARSURFACE mixedPhRateNearSurface;
    L3DPR_G1_PRECIPWATERINTEGRATED precipWaterIntegrated;
    L3DPR_G1_PRECIPICEINTEGRATED precipIceIntegrated;
    L3DPR_G1_PRECIPRAVE24 precipRateAve24;
    L3DPR_G1_ZFACTORCORRECTED zFactorCorrected;
    L3DPR_G1_ZFACTORCORRECTEDESURFACE zFactorCorrectedESurface;
    L3DPR_G1_ZFACTORCORRECTEDNEARSURFACE zFactorCorrectedNearSurface;
    L3DPR_G1_ZFACTORMEASUREDNEARSURFACE zFactorMeasuredNearSurface;
    L3DPR_G1_ZFACTORCORRECTEDDPR zFactorCorrectedDPR;
    L3DPR_G1_ZFACTORCORRECTEDESURFACEDPR zFactorCorrectedESurfaceDPR;
    L3DPR_G1_ZFACTORCORRECTEDNEARSURFACEDPR zFactorCorrectedNearSurfaceDPR;
    L3DPR_G1_ZFACTORMEASURED zFactorMeasured;
    L3DPR_G1_DM dm;
    L3DPR_G1_DBNW dBNw;
    L3DPR_G1_EPSILONDPR epsilonDPR;
    L3DPR_G1_EPSILON epsilon;
    L3DPR_G1_PIAHB piaHB;
    L3DPR_G1_PIAHYBRID piaHybrid;
    L3DPR_G1_PIAHYBRIDDPR piaHybridDPR;
}
L3DPR_G1_PIASRT piaSRT;
L3DPR_G1_PIASRTDPR piaSRTdpr;
L3DPR_G1_PIAFINAL piaFinal;
L3DPR_G1_PIAFINALDPR piaFinalDPR;
L3DPR_G1_PIAFINALSUBSET piaFinalSubset;
L3DPR_G1_PIAFINALDPRSUBSET piaFinalDPRsubset;
L3DPR_G1_HEIGHTBB heightBB;
L3DPR_G1_HEIGHTBBNADIR heightBBnadir;
L3DPR_G1_BBWIDTHNADIR BBwidthNadir;
L3DPR_G1_HEIGHTSTORMTOP heightStormTop;
L3DPR_G1_BBWIDTH BBwidth;
L3DPR_G1_OBSERVATIONCOUNTS observationCounts;
L3DPR_G1_PRECIPRATERELOCALTIME precipRateLocalTime;
L3DPR_G1_DFRMNNEARSSURFACE DFRmNearSurface;
L3DPR_G1_DFRMNEARSSURFACE DFRNearSurface;
float precipRateNearSurfaceUnconditional[5][72][28];
float precipProbabilityNearSurface[5][72][28];
}
L3DPR_G1;

#endif

ifndef _L3DPR_GRIDS_
define _L3DPR_GRIDS_
typedef struct {
    L3DPR_G1 G1;
    L3DPR_G2 G2;
} L3DPR_GRIDS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L3DPR_G2_DFRMNEARSSURFACE/
    INTEGER*4 count(536,1440,3)
    REAL*4 mean(536,1440,3)
    REAL*4 stdev(536,1440,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_DFRMNEARSSURFACE/
    INTEGER*4 count(536,1440,3)
REAL*4 mean(536,1440,3)
REAL*4 stdev(536,1440,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_OBSERVATIONCOUNTS/
   INTEGER*4 total(536,1440,4)
   INTEGER*4 pia(536,1440,4,7)
   INTEGER*4 shallowRain(536,1440,4)
END STRUCTURE

STRUCTURE /L3DPR_G2_BBWIDTH/
   INTEGER*4 count(536,1440,5,3)
   REAL*4 mean(536,1440,5,3)
   REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_HEIGHTSTORMTOP/
   INTEGER*4 count(536,1440,5,3)
   REAL*4 mean(536,1440,5,3)
   REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_HEIGHTBB/
   INTEGER*4 count(536,1440,5,3)
   REAL*4 mean(536,1440,5,3)
   REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PIAFINALDPR/
   INTEGER*4 count(536,1440,4,7,3)
   REAL*4 mean(536,1440,4,7,3)
   REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PIAFINAL/
   INTEGER*4 count(536,1440,4,7,3)
   REAL*4 mean(536,1440,4,7,3)
   REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PIASRTDPR/
   INTEGER*4 count(536,1440,4,7,3)
   REAL*4 mean(536,1440,4,7,3)
REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PIASRT/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PIAHYBRIDDPR/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PIAHYBRID/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PIAHB/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_ZETA/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_EPSILON/
    INTEGER*4 count(536,1440,4,3)
    REAL*4 mean(536,1440,4,3)
    REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_EPSILONDPR/
    INTEGER*4 count(536,1440,4,5,3)
    REAL*4 mean(536,1440,4,5,3)
    REAL*4 stdev(536,1440,4,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_DBNW/
    INTEGER*4 count(536,1440,5,3)
    REAL*4 mean(536,1440,5,3)
    REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_DM/
    INTEGER*4 count(536,1440,5,3)
    REAL*4 mean(536,1440,5,3)
    REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_ZFACTORMEASURED/
    INTEGER*4 count(536,1440,4,5,3)
    REAL*4 mean(536,1440,4,5,3)
    REAL*4 stdev(536,1440,4,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_ZFACTORCORRECTEDNEARSURFACE/DR/
    INTEGER*4 count(536,1440,4,3)
    REAL*4 mean(536,1440,4,3)
    REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_ZFACTORCORRECTEDDESIREDPR/
    INTEGER*4 count(536,1440,4,3)
    REAL*4 mean(536,1440,4,3)
    REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_ZFACTORCORRECTEDSURFACEDPR/
    INTEGER*4 count(536,1440,4,5,3)
    REAL*4 mean(536,1440,4,5,3)
    REAL*4 stdev(536,1440,4,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_ZFACTORMEASUREDEDESIREDPR/
    INTEGER*4 count(536,1440,3)
    REAL*4 mean(536,1440,3)
    REAL*4 stdev(536,1440,3)
END STRUCTURE
STRUCTURE /L3DPR_G2_ZFACTORCORRECTEDNEARSURFACE/
   INTEGER*4 count(536,1440,4,3)
   REAL*4 mean(536,1440,4,3)
   REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_ZFACTORCORRECTEDDESURFACE/
   INTEGER*4 count(536,1440,4,3)
   REAL*4 mean(536,1440,4,3)
   REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_ZFACTORCORRECTED/
   INTEGER*4 count(536,1440,4,5,3)
   REAL*4 mean(536,1440,4,5,3)
   REAL*4 stdev(536,1440,4,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PRECIPRATEAVE24/
   INTEGER*4 count(536,1440,5,3)
   REAL*4 mean(536,1440,5,3)
   REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PRECIPICEINTEGRATED/
   INTEGER*4 count(536,1440,5,3)
   REAL*4 mean(536,1440,5,3)
   REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PRECPWATERINTEGRATED/
   INTEGER*4 count(536,1440,5,3)
   REAL*4 mean(536,1440,5,3)
   REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_MIXEDPHRATENEARSURFACE/
   INTEGER*4 count(536,1440,5,3)
   REAL*4 mean(536,1440,5,3)
   REAL*4 stdev(536,1440,5,3)
END STRUCTURE
STRUCTURE /L3DPR_G2_SNOWRATENEARSURFACE/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_RAINRATENEARSURFACE/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PRECIPRATENEARSURFACE/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PRECIPRATEESURFACE2/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_MIXEDPHRATE/
  INTEGER*4 count(536,1440,5,5,3)
  REAL*4 mean(536,1440,5,5,3)
  REAL*4 stdev(536,1440,5,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_FLAGHEAVYICEPRECIP/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_SNOWRATE/
INTEGER*4 count(536,1440,5,5,3)
REAL*4 mean(536,1440,5,5,3)
REAL*4 stdev(536,1440,5,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_RAINRATE/
  INTEGER*4 count(536,1440,5,5,3)
  REAL*4 mean(536,1440,5,5,3)
  REAL*4 stdev(536,1440,5,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2_PRECIPRATE/
  INTEGER*4 count(536,1440,5,5,3)
  REAL*4 mean(536,1440,5,5,3)
  REAL*4 stdev(536,1440,5,5,3)
END STRUCTURE

STRUCTURE /L3DPR_G2/
  RECORD /L3DPR_G2_PRECIPRATE/ precipRate
  RECORD /L3DPR_G2_RAINRATE/ rainRate
  RECORD /L3DPR_G2_SNOWRATE/ snowRate
  RECORD /L3DPR_G2_FLAGHEAVYICEPRECIP/ flagHeavyIcePrecip
  RECORD /L3DPR_G2_MIXEDPHRATE/ mixedPhRate
  RECORD /L3DPR_G2_PRECIPRATESURFACE/ precipRateESurface
  RECORD /L3DPR_G2_PRECIPRATESURFACE2/ precipRateESurface2
  RECORD /L3DPR_G2_PRECIPRATENEARSURFACE/ precipRateNearSurface
  RECORD /L3DPR_G2_RAINRATENEARSURFACE/ rainRateNearSurface
  RECORD /L3DPR_G2_SNOWRATENEARSURFACE/ snowRateNearSurface
  RECORD /L3DPR_G2_MIXEDPHRATENEARSURFACE/ mixedPhRateNearSurface
  RECORD /L3DPR_G2_PRECIPWATERINTEGRATED/ precipWaterIntegrated
  RECORD /L3DPR_G2_PRECIPICEINTEGRATED/ precipIceIntegrated
  RECORD /L3DPR_G2_PRECIPRATAVE24/ precipRateAve24
  RECORD /L3DPR_G2_ZFACTORCORRECTED/ zFactorCorrected
  RECORD /L3DPR_G2_ZFACTORCORRECTEDESURFACE/ zFactorCorrectedESurface
  RECORD /L3DPR_G2_ZFACTORCORRECTEDNEARSURFACE/ zFactorCorrectedNearSurface
  RECORD /L3DPR_G2_ZFACTORMEASUREDNEARSURFACE/ zFactorMeasuredNearSurface
  RECORD /L3DPR_G2_ZFACTORCORRECTEDDPR/ zFactorCorrectedDPR
  RECORD /L3DPR_G2_ZFACTORCORRECTEDESURFACE_DPR/ zFactorCorrectedESurfaceDPR
  RECORD /L3DPR_G2_ZFACTORMEASURED/ zFactorMeasured
  RECORD /L3DPR_G2_DM/ dm
  RECORD /L3DPR_G2_DBNW/ DBNW
  RECORD /L3DPR_G2_EPSILON_DPR/ epsilonDPR
RECORD /L3DPR_G2_EPSILON/ epsilon
RECORD /L3DPR_G2_ZETA/ zeta
RECORD /L3DPR_G2_PIAHB/ piaHB
RECORD /L3DPR_G2_PIAHYBRID/ piaHybrid
RECORD /L3DPR_G2_PIAHYBRIDDPR/ piaHybridDPR
RECORD /L3DPR_G2_PIASRT/ piaSRT
RECORD /L3DPR_G2_PIASRTDPR/ piaSRTdpr
RECORD /L3DPR_G2_PIAFINAL/ piaFinal
RECORD /L3DPR_G2_PIAFINALDPR/ piaFinalDPR
RECORD /L3DPR_G2_HEIGHTBB/ heightBB
RECORD /L3DPR_G2_HEIGHTSTORMTOP/ heightStormTop
RECORD /L3DPR_G2_BBWIDTH/ BBwidth
RECORD /L3DPR_G2_ObservationCounts/ observationCounts
RECORD /L3DPR_G2_DFRMNearsurface/ DFRmNearSurface
RECORD /L3DPR_G2_DFRNearsurface/ DFRNearSurface
REAL*4 precipRateNearSurfaceUnconditional(536,1440,5)
REAL*4 precipProbabilityNearSurface(536,1440,5)
END STRUCTURE

STRUCTURE /L3DPR_G1_DFRNearsurface/
  INTEGER*4 count(28,72,3,3)
  REAL*4 mean(28,72,3,3)
  REAL*4 stdev(28,72,3,3)
  INTEGER*4 hist(28,72,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_DFRMnearsurface/
  INTEGER*4 count(28,72,3,3)
  REAL*4 mean(28,72,3,3)
  REAL*4 stdev(28,72,3,3)
  INTEGER*4 hist(28,72,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PrecipRateLocalTime/
  INTEGER*4 count(28,72,5,24,3)
  REAL*4 mean(28,72,5,24,3)
  REAL*4 stdev(28,72,5,24,3)
END STRUCTURE

STRUCTURE /L3DPR_G1_ObservationCounts/
  INTEGER*4 total(28,72,4,3)
  INTEGER*4 localTime(28,72,4,24,3)
  INTEGER*4 pia(28,72,4,7,3)
INTEGER*4 shallowRain(28,72,4,3)
END STRUCTURE

STRUCTURE /L3DPR_G1_BBWIDTH/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_HEIGHTSTORMTOP/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_BBWIDTHNADIR/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_HEIGHTBBNADIR/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_HEIGHTBB/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PIAFINALDPRSUBSET/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PIAFINALSUBSET/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PIAFINALDPR/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PIAFINAL/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PIASRTDPR/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PIASRT/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PIASHYBRIDDPR/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE
STRUCTURE /L3DPR_G1_PIAHYBRID/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PIAHB/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_ZETA/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_EPSILON/
  INTEGER*4 count(28,72,4,3,3)
  REAL*4 mean(28,72,4,3,3)
  REAL*4 stdev(28,72,4,3,3)
  INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_EPSILONDPR/
  INTEGER*4 count(28,72,4,5,3,3)
  REAL*4 mean(28,72,4,5,3,3)
  REAL*4 stdev(28,72,4,5,3,3)
  INTEGER*4 hist(28,72,4,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_DBNW/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE
STRUCTURE /L3DPR_G1_DM/
    INTEGER*4 count(28,72,5,3,3)
    REAL*4 mean(28,72,5,3,3)
    REAL*4 stdev(28,72,5,3,3)
    INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_ZFACTORMEASURED/
    INTEGER*4 count(28,72,4,5,3,3)
    REAL*4 mean(28,72,4,5,3,3)
    REAL*4 stdev(28,72,4,5,3,3)
    INTEGER*4 hist(28,72,4,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_ZFACTORCORRECTEDNEARSURFACEDPR/
    INTEGER*4 count(28,72,4,3,3)
    REAL*4 mean(28,72,4,3,3)
    REAL*4 stdev(28,72,4,3,3)
    INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_ZFACTORCORRECTEDDESURFACEDPR/
    INTEGER*4 count(28,72,4,3,3)
    REAL*4 mean(28,72,4,3,3)
    REAL*4 stdev(28,72,4,3,3)
    INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_ZFACTORCORRECTEDDPR/
    INTEGER*4 count(28,72,4,5,3,3)
    REAL*4 mean(28,72,4,5,3,3)
    REAL*4 stdev(28,72,4,5,3,3)
    INTEGER*4 hist(28,72,4,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_ZFACTORMEASUREDNEARSURFACESPR/
    INTEGER*4 count(28,72,4,3,3)
    REAL*4 mean(28,72,4,3,3)
    REAL*4 stdev(28,72,4,3,3)
    INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_ZFACTORCORRECTEDNEARSURFACESPR/
    INTEGER*4 count(28,72,4,3,3)
    REAL*4 mean(28,72,4,3,3)
    REAL*4 stdev(28,72,4,3,3)
    INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_ZFACTORCORRECTEDNEARSURFACESPR/
INTEGER*4 count(28,72,4,3,3)
REAL*4 mean(28,72,4,3,3)
REAL*4 stdev(28,72,4,3,3)
INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_ZFACTORCORRECTED/;
    INTEGER*4 count(28,72,4,5,3,3)
    REAL*4 mean(28,72,4,5,3,3)
    REAL*4 stdev(28,72,4,5,3,3)
    INTEGER*4 hist(28,72,4,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PRECIPRATEAVE24/;
    INTEGER*4 count(28,72,5,3,3)
    REAL*4 mean(28,72,5,3,3)
    REAL*4 stdev(28,72,5,3,3)
    INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PRECIPICEINTEGRATED/;
    INTEGER*4 count(28,72,5,3,3)
    REAL*4 mean(28,72,5,3,3)
    REAL*4 stdev(28,72,5,3,3)
    INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_MIXEDPHRATENEARSURFACE/;
    INTEGER*4 count(28,72,5,3,3)
REAL*4 mean(28,72,5,3,3)
REAL*4 stdev(28,72,5,3,3)
INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_SNOWRATENEARSURFACE/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_RAINRATENEARSURFACE/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PRECIPRATENEARSURFACE/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_MIXEDPHRATE/
  INTEGER*4 count(28,72,5,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PRECIPRATEESURFACE2/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PRECIPRATEESURFACE/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_MIXEDPHRATE/
  INTEGER*4 count(28,72,5,5,3,3)
  REAL*4 mean(28,72,5,3,3)
REAL*4 stdev(28,72,5,5,3,3)
INTEGER*4 hist(28,72,5,5,3,3)
END STRUCTURE

STRUCTURE /L3DPR_G1_FLAGHEAVYICEPRECIP/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_SNOWRATE/
  INTEGER*4 count(28,72,5,5,3,3)
  REAL*4 mean(28,72,5,5,3,3)
  REAL*4 stdev(28,72,5,5,3,3)
  INTEGER*4 hist(28,72,5,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_RAINRATE/
  INTEGER*4 count(28,72,5,5,3,3)
  REAL*4 mean(28,72,5,5,3,3)
  REAL*4 stdev(28,72,5,5,3,3)
  INTEGER*4 hist(28,72,5,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1_PRECIPRATE/
  INTEGER*4 count(28,72,5,5,3,3)
  REAL*4 mean(28,72,5,5,3,3)
  REAL*4 stdev(28,72,5,5,3,3)
  INTEGER*4 hist(28,72,5,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPR_G1/
  RECORD /L3DPR_G1_PRECIPRATE/ precipRate
  RECORD /L3DPR_G1_RAINRATE/ rainRate
  RECORD /L3DPR_G1_SNOWRATE/ snowRate
  RECORD /L3DPR_G1_FLAGHEAVYICEPRECIP/ flagHeavyIcePrecip
  RECORD /L3DPR_G1_MIXEDPHRATE/ mixedPhRate
  RECORD /L3DPR_G1_PRECIPRATEESURFACE/ precipRateESurface
  RECORD /L3DPR_G1_PRECIPRATEESURFACE2/ precipRateESurface2
  RECORD /L3DPR_G1_PRECIPRATENEARSURFACE/ precipRateNearSurface
  RECORD /L3DPR_G1_RAINRATENEARSURFACE/ rainRateNearSurface
  RECORD /L3DPR_G1_SNOWRATENEARSURFACE/ snowRateNearSurface
RECORD /L3DPR_G1_MIXEDPHRATENEARSURFACE/ mixedPhRateNearSurface
RECORD /L3DPR_G1_PRECIPWATERINTEGRATED/ precipWaterIntegrated
RECORD /L3DPR_G1_PRECIPICEINTEGRATED/ precipIceIntegrated
RECORD /L3DPR_G1_PRECIPRATEAVE24/ precipRateAve24
RECORD /L3DPR_G1_ZFACTORCORRECTED/ zFactorCorrected
RECORD /L3DPR_G1_ZFACTORCORRECTEDESURFACE/ zFactorCorrectedESurface
RECORD /L3DPR_G1_ZFACTORCORRECTEDNEARSURFACE/ zFactorCorrectedNearSurface
RECORD /L3DPR_G1_ZFACTORMEASUREDNEARSURFACE/ zFactorMeasuredNearSurface
RECORD /L3DPR_G1_ZFACTORCORRECTEDDPR/ zFactorCorrectedDPR
RECORD /L3DPR_G1_ZFACTORCORRECTEDESURFACEDPR/ zFactorCorrectedESurfaceDPR
RECORD /L3DPR_G1_ZFACTORCORRECTEDNEARSURFACEDPR/ zFactorCorrectedNearSurfaceDPR
RECORD /L3DPR_G1_ZFACTORMEASURED/ zFactorMeasured
RECORD /L3DPR_G1_DM/ dm
RECORD /L3DPR_G1_DBNW/ dBNw
RECORD /L3DPR_G1_EPSILON/ epsilon
RECORD /L3DPR_G1_EPSILONDPR/ epsilonDPR
RECORD /L3DPR_G1_ZETA/ zeta
RECORD /L3DPR_G1_PIAHB/ piaHB
RECORD /L3DPR_G1_PIAHYBRID/ piaHybrid
RECORD /L3DPR_G1_PIAHYBRIDDPR/ piaHybridDPR
RECORD /L3DPR_G1 PIASRT/ piaSRT
RECORD /L3DPR_G1_PIASRTDPR/ piaSRTdpr
RECORD /L3DPR_G1_PIAFINAL/ piaFinal
RECORD /L3DPR_G1_PIAFINALDPR/ piaFinalDPR
RECORD /L3DPR_G1_PIAFINALSUBSET/ piaFinalSubset
RECORD /L3DPR_G1_PIAFINALDPRSUBSET/ piaFinalDPRsubset
RECORD /L3DPR_G1_HEIGHTBB/ heightBB
RECORD /L3DPR_G1_HEIGHTBBNADIR/ heightBBnadir
RECORD /L3DPR_G1_BBWIDTHNADIR/ BBwidthNadir
RECORD /L3DPR_G1_HEIGHTSTORMTOP/ heightStormTop
RECORD /L3DPR_G1_BBWIDTH/ BBwidth
RECORD /L3DPR_G1_OBSERVATIONCOUNTS/ observationCounts
RECORD /L3DPR_G1_PRECIPRATELOCALTIME/ precipRateLocalTime
RECORD /L3DPR_G1_DFRMNEARSURFACE/ DFRmNearSurface
RECORD /L3DPR_G1_DFRNEARSURFACE/ DFRNearSurface
REAL*4 precipRateNearSurfaceUnconditional(28,72,5)
REAL*4 precipProbabilityNearSurface(28,72,5)
END STRUCTURE

STRUCTURE /L3DPR_GRIDS/
  RECORD /L3DPR_G1/ G1
  RECORD /L3DPR_G2/ G2
END STRUCTURE
3DPRD - DPR Daily Product

3DPRD, "DPR Daily Product", computes daily statistics of the DPR measurements at a high horizontal resolution (0.25° x 0.25° latitude/longitude).

Dimension definitions:

nlat 536 Number of high resolution 0.25° grid intervals of latitude from 67°S to 67°N.
nlon 1440 Number of high resolution 0.25° grid intervals of longitude from 180°W to 180°E.
nalt 5 Number of heights above the earth ellipsoid: 2km, 4km, 6km, 10km, and 15km.
nvar 3 Number of phase bins. Bins are counts of phase less than 100, counts of phase greater than or equal to 100 and less than 200, counts of phase greater than or equal to 200.
chn 2 Number of channels. Channels are first Ku and second DPR.
AD 2 Ascending or descending half of the orbit.

Figure 692 through Figure 695 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.
Figure 692: Data Format Structure for 3DPRD, DPR Daily Product
continued from last figure

**GRID**

- **convPrecipRateESurfMean** 4 bytes  Array: nlat x nlon x chn x AD
- **convPrecipPixNearSurf** 2 bytes  Array: nlat x nlon x chn x AD
- **stratPrecipRateMean** 4 bytes  Array: nlat x nlon x chn x AD
- **stratPrecipRateNearSurfMean** 4 bytes  Array: nlat x nlon x chn x AD
- **stratPrecipRateESurfMean** 4 bytes  Array: nlat x nlon x chn x AD
- **stratPrecipPixNearSurf** 2 bytes  Array: nlat x nlon x chn x AD
- **bbHtMean** 4 bytes  Array: nlat x nlon x chn x AD
- **stormHtMean** 4 bytes  Array: nlat x nlon x chn x AD
- **phase** 2 bytes  Array: nlat x nlon x nalt x nvar x chn x AD
- **phaseNearSurf** 2 bytes  Array: nlat x nlon x nvar x chn x AD
- **GridTimeAsc**  Group
- **GridTimeDes**  Group

Figure 693: Data Format Structure for 3DPRD, DPR Daily Product

**GridTimeAsc**

- **Year** 2 bytes  Array: nlat x nlon
- **Month** 1 byte  Array: nlat x nlon
- **DayOfMonth** 1 byte  Array: nlat x nlon
- **Hour** 1 byte  Array: nlat x nlon
- **Minute** 1 byte  Array: nlat x nlon
- **Second** 1 byte  Array: nlat x nlon
- **MilliSecond** 2 bytes  Array: nlat x nlon
- **DayOfYear** 2 bytes  Array: nlat x nlon

Figure 694: Data Format Structure for 3DPRD, GridTimeAsc
GRID (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
Mean Precipitation rate, includes both liquid and solid phases at various height levels. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

rainRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
Mean rainfall rate, excludes solid precipitation at various height levels. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

mixedRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
Mean rainfall rate of the mixed phase precipitation at various height levels. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

snowRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
Mean rainfall rate of solid precipitation at various height levels. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateNearSurfMean (4-byte float, array size: nlat x nlon x chn x AD):
Mean precipitation rate in a grid box using only the Near Surface location along the slant path for each radar ray. First index is Ascending node, second index is Descending.
Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

rainRateNearSurfMean (4-byte float, array size: nlat x nlon x chn x AD):
Mean rainfall rate of liquid precipitation in a grid box using only the Near Surface location
along the slant path for each radar ray. First index is Ascending node, second index is
Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

mixedRateNearSurfMean (4-byte float, array size: nlat x nlon x chn x AD):
Mean rainfall rate of mixed phase precipitation in a grid box using only the Near Surface
location along the slant path for each radar ray. First index is Ascending node, second
index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

snowRateNearSurfMean (4-byte float, array size: nlat x nlon x chn x AD):
Mean rainfall rate of solid precipitation in a grid box using only the Near Surface location
along the slant path for each radar ray. First index is Ascending node, second index is
Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateESurfMean (4-byte float, array size: nlat x nlon x chn x AD):
Mean precipitation rate in a grid box using only the Estimated Surface location along the
slant path for each radar ray. First index is Ascending node, second index is Descending.
Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateESurf2Mean (4-byte float, array size: nlat x nlon x chn x AD):
Mean precipitation rate in a grid box using only the Estimated Surface 2 location along the
slant path for each radar ray. First index is Ascending node, second index is Descending.
Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

totalPix (2-byte integer, array size: nlat x nlon x chn x AD):
The total number of measurements in each grid box. First index is Ascending node,
second index is Descending. Special values are defined as:
-9999 Missing value

precipPix (2-byte integer, array size: nlat x nlon x nalt x chn x AD):
The number of measurements in each grid box that included detectable precipitation at
various height levels. First index is Ascending node, second index is Descending. Special
values are defined as:
-9999 Missing value

precipPixNearSurf (2-byte integer, array size: nlat x nlon x chn x AD):
The number of measurements in a grid box that included detectable precipitation at the
Near Surface level. First index is Ascending node, second index is Descending. Special
values are defined as:
-9999 Missing value
precipPixESurf (2-byte integer, array size: nlat x nlon x chn x AD):
The number of measurements in a grid box that included detectable precipitation at
the Estimated Surface level. First index is Ascending node, second index is Descending.
Special values are defined as:
-9999  Missing value

convPrecipRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
The mean precipitation rate of convective type at various height levels. First index is
Ascending node, second index is Descending. Values are in mm/hr. Special values are
defined as:
-9999.9  Missing value

convPrecipRateNearSurfMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean precipitation rate of convective type at the Near Surface level along the radar
ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are
defined as:
-9999.9  Missing value

convPrecipRateESurfMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean precipitation rate of convective type at the Estimated Surface level along the
radar ray. First index is Ascending node, second index is Descending. Values are in
mm/hr. Special values are defined as:
-9999.9  Missing value

convPrecipPixNearSurf (2-byte integer, array size: nlat x nlon x chn x AD):
The number of convective precipitation measurements in a grid box at the Near Surface
level. First index is Ascending node, second index is Descending. Special values are
defined as:
-9999  Missing value

stratPrecipRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
The mean precipitation rate of stratiform type at various height levels. First index is
Ascending node, second index is Descending. Values are in mm/hr. Special values are
defined as:
-9999.9  Missing value

stratPrecipRateNearSurfMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean precipitation rate of stratiform type at the Near Surface level along the radar
ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are
defined as:
-9999.9  Missing value

stratPrecipRateESurfMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean precipitation rate of stratiform type at the Estimated Surface level along the
radar ray. First index is Ascending node, second index is Descending. Values are in
mm/hr. Special values are defined as:
-9999.9  Missing value
stratPrecipPixNearSurf (2-byte integer, array size: nlat x nlon x chn x AD):
The number of stratiform precipitation measurements in a grid box at the Near Surface
level. First index is Ascending node, second index is Descending. Special values are
declared as:
   -9999  Missing value

bbHtMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean bright band height in a grid box. First index is Ascending node, second index
is Descending. Values are in m. Special values are defined as:
   -9999.9  Missing value

stormHtMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean storm height in a grid box. First index is Ascending node, second index is
Descending. Values are in m. Special values are defined as:
   -9999.9  Missing value

phase (2-byte integer, array size: nlat x nlon x nalt x nvar x chn x AD):
The precipitation phase type in a grid box at various heights. First index is Ascending
node, second index is Descending. Special values are defined as:
   -9999  Missing value

phaseNearSurf (2-byte integer, array size: nlat x nlon x nvar x chn x AD):
The precipitation phase type in a grid box. First index is Ascending node, second index
is Descending. Special values are defined as:
   -9999  Missing value

GridTimeAsc (Group)
A UTC time associated with the grid box.

Year (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined
as:
   -9999  Missing value

Month (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
   -99  Missing value

DayOfMonth (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
   -99  Missing value

Hour (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
   -99  Missing value
Minute (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

GridTimeDes (Group)
A UTC time associated with the grid box.

Year (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

Month (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value
DayOfYear (2-byte integer, array size: nlat x nlon): Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

C Structure Header file:

```c
#ifndef _TK_3DPRD_H_
#define _TK_3DPRD_H_

#ifndef _L3DPRD_GRIDTIMEDES_
#define _L3DPRD_GRIDTIMEDES_

typedef struct {
    short Year[1440][536];
    signed char Month[1440][536];
    signed char DayOfMonth[1440][536];
    signed char Hour[1440][536];
    signed char Minute[1440][536];
    signed char Second[1440][536];
    short MilliSecond[1440][536];
    short DayOfYear[1440][536];
} L3DPRD_GRIDTIMEDES;

#endif

#ifndef _L3DPRD_GRIDTIMEASC_
#define _L3DPRD_GRIDTIMEASC_

typedef struct {
    short Year[1440][536];
    signed char Month[1440][536];
    signed char DayOfMonth[1440][536];
    signed char Hour[1440][536];
    signed char Minute[1440][536];
    signed char Second[1440][536];
    short MilliSecond[1440][536];
    short DayOfYear[1440][536];
} L3DPRD_GRIDTIMEASC;

#endif

#ifndef _L3DPRD_GRID_
#define _L3DPRD_GRID_

#endif
```

typedef struct {
    float precipRateMean[2][2][5][1440][536];
    float rainRateMean[2][2][5][1440][536];
    float mixedRateMean[2][2][5][1440][536];
    float snowRateMean[2][2][5][1440][536];
    float precipRateNearSurfMean[2][2][1440][536];
    float rainRateNearSurfMean[2][2][1440][536];
    float mixedRateNearSurfMean[2][2][1440][536];
    float snowRateNearSurfMean[2][2][1440][536];
    float precipRateESurfMean[2][2][1440][536];
    float precipRateESurf2Mean[2][2][1440][536];
    short totalPix[2][2][1440][536];
    short precipPix[2][2][5][1440][536];
    short precipPixNearSurf[2][2][1440][536];
    short precipPixESurf[2][2][1440][536];
    float convPrecipRateMean[2][2][5][1440][536];
    float convPrecipRateNearSurfMean[2][2][1440][536];
    float convPrecipRateESurfMean[2][2][1440][536];
    short convPrecipPixNearSurf[2][2][1440][536];
    float stratPrecipRateMean[2][2][5][1440][536];
    float stratPrecipRateNearSurfMean[2][2][1440][536];
    float stratPrecipRateESurfMean[2][2][1440][536];
    short stratPrecipPixNearSurf[2][2][1440][536];
    float bbHtMean[2][2][1440][536];
    float stormHtMean[2][2][1440][536];
    short phase[2][2][3][5][1440][536];
    short phaseNearSurf[2][2][3][1440][536];
    L3DPRD_GRIDTIMEASC GridTimeAsc;
    L3DPRD_GRIDTIMEDES GridTimeDes;
} L3DPRD_GRID;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L3DPRD_GRIDTIMEDES/
    INTEGER*2 Year(536,1440)
    BYTE Month(536,1440)
    BYTE DayOfMonth(536,1440)
    BYTE Hour(536,1440)
    BYTE Minute(536,1440)
BYTE Second(536,1440)
INTEGER*2 MilliSecond(536,1440)
INTEGER*2 DayOfYear(536,1440)
END STRUCTURE

STRUCTURE /L3DPRD_GRIDTIMEASC/
  INTEGER*2 Year(536,1440)
  BYTE Month(536,1440)
  BYTE DayOfMonth(536,1440)
  BYTE Hour(536,1440)
  BYTE Minute(536,1440)
  BYTE Second(536,1440)
  INTEGER*2 MilliSecond(536,1440)
  INTEGER*2 DayOfYear(536,1440)
END STRUCTURE

STRUCTURE /L3DPRD_GRID/
  REAL*4 precipRateMean(536,1440,5,2,2)
  REAL*4 rainRateMean(536,1440,5,2,2)
  REAL*4 mixedRateMean(536,1440,5,2,2)
  REAL*4 snowRateMean(536,1440,5,2,2)
  REAL*4 precipRateNearSurfMean(536,1440,2,2)
  REAL*4 rainRateNearSurfMean(536,1440,2,2)
  REAL*4 mixedRateNearSurfMean(536,1440,2,2)
  REAL*4 snowRateNearSurfMean(536,1440,2,2)
  REAL*4 precipRateESurfMean(536,1440,2,2)
  REAL*4 precipRateESurf2Mean(536,1440,2,2)
  INTEGER*2 totalPix(536,1440,2,2)
  INTEGER*2 precipPix(536,1440,5,2,2)
  INTEGER*2 precipPixNearSurf(536,1440,2,2)
  INTEGER*2 precipPixESurf(536,1440,2,2)
  REAL*4 convPrecipRateMean(536,1440,5,2,2)
  REAL*4 convPrecipRateNearSurfMean(536,1440,2,2)
  REAL*4 convPrecipRateESurfMean(536,1440,2,2)
  INTEGER*2 convPrecipPixNearSurf(536,1440,2,2)
  REAL*4 stratPrecipRateMean(536,1440,5,2,2)
  REAL*4 stratPrecipRateNearSurfMean(536,1440,2,2)
  REAL*4 stratPrecipRateESurfMean(536,1440,2,2)
  INTEGER*2 stratPrecipPixNearSurf(536,1440,2,2)
  REAL*4 bbHtMean(536,1440,2,2)
  REAL*4 stormHtMean(536,1440,2,2)
  INTEGER*2 phase(536,1440,5,3,2,2)
  INTEGER*2 phaseNearSurf(536,1440,3,2,2)
3PR, "PR Full Product", computes statistics of the PR measurements at both a low horizontal resolution (G1, 5° x 5° latitude/longitude) and a high horizontal resolution (G2, 0.25° x 0.25° latitude/longitude). The product can be monthly or daily.

Histograms have the following category thresholds, where
histbin(i) = cat(i) less than x less than or equal to cat(i+1)

- cat rain = [ 0.01, ! mm/h (logarithmic steps)
  0.10, 0.13, 0.17, 0.23, 0.30, 0.40,
  0.52, 0.69, 0.91, 1.20, 1.58, 2.08,
  2.75, 3.62, 4.77, 6.29, 8.29, 10.92,
  14.40, 18.97, 25.00, 32.95, 43.43, 57.24,
  75.44, 99.43, 131.04, 172.71, 227.63, 300.00 ],

- cat Z = [ 0.01, ! dBZ
  6.0, 8.0, 10.0, 12.0, 14.0, 16.0,
  18.0, 20.0, 22.0, 24.0, 26.0, 28.0,
  30.0, 32.0, 34.0, 36.0, 38.0, 40.0,
  42.0, 44.0, 46.0, 48.0, 50.0, 52.0,
  54.0, 56.0, 58.0, 60.0, 62.0, 64.0 ],

- cat integratedWater = [ 0.0, ! kg/m²
  200.0, 400.0, 600.0, 800.0, 1000.0, 1200.0,
  1400.0, 1600.0, 1800.0, 2000.0, 2200.0, 2400.0,
  2600.0, 2800.0, 3000.0, 3200.0, 3400.0, 3600.0,
  3800.0, 4000.0, 4200.0, 4400.0, 4600.0, 4800.0,
  5000.0, 5200.0, 5400.0, 5600.0, 5800.0, 6000.0 ],

- cat bbhgt = [ 10.0, ! meters
  250.0, 500.0, 750.0, 1000.0, 1250.0, 1500.0,
  1750.0, 2000.0, 2250.0, 2500.0, 2750.0, 3000.0,
  3250.0, 3500.0, 3750.0, 4000.0, 4250.0, 4500.0,
  4750.0, 5000.0, 5250.0, 5500.0, 5750.0, 6000.0,
  6250.0, 6500.0, 6750.0, 7000.0, 7500.0, 20000.0 ],

- cat bbwdth = [ 0.0, ! meters
125.0, 250.0, 375.0, 500.0, 625.0, 750.0, 875.0, 1000.0, 1125.0, 1250.0, 1375.0, 1500.0, 1625.0, 1750.0, 1875.0, 2000.0, 2125.0, 2250.0, 2375.0, 2500.0, 2625.0, 2750.0, 2875.0, 3000.0, 3125.0, 3250.0, 3375.0, 3500.0, 3625.0, 3750.0 ]

cat stormh = 1000.0*[ 0.01, ! km (convert m > km) 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0, 8.5, 9.0, 9.5, 10.0, 10.5, 11.0, 11.5, 12.0, 12.5, 13.0, 14.0, 15.0, 16.0, 20.0 ]

cat epsilon = [ 0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0 ]

cat nubf = [ 1.0, 1.05, 1.1, 1.15, 1.2, 1.25, 1.3, 1.35, 1.4, 1.45, 1.5, 1.55, 1.6, 1.65, 1.7, 1.75, 1.8, 1.85, 1.9, 1.95, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0 ]

cat pia = [ 0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.8, 1.0, 1.2, 1.4, 1.6, 1.8, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 7.0, 8.0, 9.0, 10.0, 15.0, 20.0, 25.0, 30.0, 100.0 ]

cat dBNw = [ 0.1, 1.0, 2.0, 4.0, 6.0, 8.0, 10.0, 12.0, 14.0, 16.0, 18.0, 20.0, 22.0, 24.0, 26.0, 28.0, 30.0, 32.0, 34.0, 36.0, 38.0, 40.0, 42.0, 44.0, 46.0, 48.0, 50.0, 52.0, 54.0, 56.0, 60.0 ]

cat Dm = [ 0.1, ! mm 0.2, 0.3, 0.4, 0.5, 0.6, 0.7,
Dimension definitions:

\[ \begin{align*}
\text{ltL} & \quad 28 \quad \text{Number of low resolution 5° grid intervals of latitude from 70°S to 70°N.} \\
\text{lnL} & \quad 72 \quad \text{Number of low resolution 5° grid intervals of longitude from 180°W to 180°E.} \\
\text{ltH} & \quad 536 \quad \text{Number of high resolution 0.25° grid intervals of latitude from 67°S to 67°N.} \\
\text{lnH} & \quad 1440 \quad \text{Number of high resolution 0.25° grid intervals of longitude from 180°W to 180°E.} \\
\text{chn} & \quad 5 \quad \text{Number of channels: Ku, Ka, KaHS, DPR, KuMS.} \\
\text{inst} & \quad 4 \quad \text{Number of instruments: Ku, Ka, KaHS, KuMS.} \\
\text{hgt} & \quad 5 \quad \text{Number of heights above the earth ellipsoid: 2, 4, 6, 10, and 15 km.} \\
\text{tim} & \quad 24 \quad \text{Number of hours (local time).} \\
\text{ang} & \quad 7 \quad \text{Number of angles. The meaning of ang is different for each channel.} \\
& \quad \text{For Ku channel all indeces are used with the meaning 0, 1, 2,...,} 6 = \text{angle bins 24, (20,28), (16,32), (12,36), (8,40), (3,44), and (0,48).} \\
& \quad \text{For Ka channel 4 indeces are used with the meaning 0, 1, 2, 3 = angle bins 12, (8,16), (4,20), and (0,24).} \\
& \quad \text{For KaHS channel 4 indeces are used with the meaning 0, 1, 2, 3 = angle bins (11,2), (7,16), (3,20), and (0,23).} \\
\text{rt} & \quad 3 \quad \text{Number of rain types: stratiform, convective, all.} \\
\text{st} & \quad 3 \quad \text{Number of surface types: ocean, land, all.} \\
\text{bin} & \quad 30 \quad \text{Number of bins in histogram. The thresholds are different for different variables. See the introduction to this algorithm.}
\end{align*} \]

Figure 696 through Figure 786 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 696: Data Format Structure for 3PR, PR Full Product
Figure 697: Data Format Structure for 3PR, G1, G1
continued from last figure

---

Figure 698: Data Format Structure for 3PR, G1, G1
continued from last figure

Figure 699: Data Format Structure for 3PR, G1

Figure 700: Data Format Structure for 3PR, G2, G2
Figure 701: Data Format Structure for 3PR, G2

Figure 702: Data Format Structure for 3PR, G1, precipRate
**Figure 703: Data Format Structure for 3PR, G1, rainRate**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x hgt x rt x st

**Figure 704: Data Format Structure for 3PR, G1, snowRate**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x hgt x rt x st

**Figure 705: Data Format Structure for 3PR, G1, flagHeavyIcePrecip**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x rt x st

**Figure 706: Data Format Structure for 3PR, G1, mixedPhRate**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x hgt x rt x st

**Figure 707: Data Format Structure for 3PR, G1, precipRateESurface**

- **count**: 4 bytes
- **mean**: 4 bytes
- **stdev**: 4 bytes
- **hist**: 4 bytes

Array: ltL x lnL x chn x rt x st x bin
Figure 708: Data Format Structure for 3PR, G1, precipRateESurface2

Figure 709: Data Format Structure for 3PR, G1, precipRateNearSurface

Figure 710: Data Format Structure for 3PR, G1, rainRateNearSurface
5.54 3PR - PR Full Product

Figure 711: Data Format Structure for 3PR, G1, snowRateNearSurface

Figure 712: Data Format Structure for 3PR, G1, mixedPhRateNearSurface

Figure 713: Data Format Structure for 3PR, G1, precipWaterIntegrated

Figure 714: Data Format Structure for 3PR, G1, precipIceIntegrated

Figure 715: Data Format Structure for 3PR, G1, precipRateAve24
Figure 716: Data Format Structure for 3PR, G1, zFactorCorrected

Figure 717: Data Format Structure for 3PR, G1, zFactorCorrectedESurface

Figure 718: Data Format Structure for 3PR, G1, zFactorCorrectedNearSurface

Figure 719: Data Format Structure for 3PR, G1, zFactorMeasuredNearSurface

Figure 720: Data Format Structure for 3PR, G1, zFactorCorrectedDPR
Figure 721: Data Format Structure for 3PR, G1, zFactorCorrectedESurfaceDPR

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

Figure 722: Data Format Structure for 3PR, G1, zFactorCorrectedNearSurfaceDPR

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

Figure 723: Data Format Structure for 3PR, G1, zFactorMeasured

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

Figure 724: Data Format Structure for 3PR, G1, dm

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

Figure 725: Data Format Structure for 3PR, G1, dBNw

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes
Figure 726: Data Format Structure for 3PR, G1, epsilonDPR

Figure 727: Data Format Structure for 3PR, G1, epsilon

Figure 728: Data Format Structure for 3PR, G1, zeta

Figure 729: Data Format Structure for 3PR, G1, piaHB

Figure 730: Data Format Structure for 3PR, G1, piaHybrid
Figure 731: Data Format Structure for 3PR, G1, piaHybridDPR

Figure 732: Data Format Structure for 3PR, G1, piaSRT

Figure 733: Data Format Structure for 3PR, G1, piaSRTdpr

Figure 734: Data Format Structure for 3PR, G1, piaFinal

Figure 735: Data Format Structure for 3PR, G1, piaFinalDPR
5 STANDARD GPM PRODUCTS

Figure 736: Data Format Structure for 3PR, G1, piaFinalSubset

Figure 737: Data Format Structure for 3PR, G1, piaFinalDPRsubset

Figure 738: Data Format Structure for 3PR, G1, heightBB

Figure 739: Data Format Structure for 3PR, G1, heightBBnadir

Figure 740: Data Format Structure for 3PR, G1, BBwidthNadir
Figure 741: Data Format Structure for 3PR, G1, heightStormTop

Figure 742: Data Format Structure for 3PR, G1, BBwidth

Figure 743: Data Format Structure for 3PR, G1, observationCounts

Figure 744: Data Format Structure for 3PR, G1, precipRateLocalTime

Figure 745: Data Format Structure for 3PR, G1, DFRmNearSurface

Figure 746: Data Format Structure for 3PR, G1, DFRNearSurface
Figure 747: Data Format Structure for 3PR, G2, precipRate

Figure 748: Data Format Structure for 3PR, G2, rainRate

Figure 749: Data Format Structure for 3PR, G2, snowRate

Figure 750: Data Format Structure for 3PR, G2, flagHeavyIcePrecip
mixedPhRate
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes

Figure 751: Data Format Structure for 3PR, G2, mixedPhRate

precipRateESurface
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes

Figure 752: Data Format Structure for 3PR, G2, precipRateESurface

precipRateESurface2
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes

Figure 753: Data Format Structure for 3PR, G2, precipRateESurface2

precipRateNearSurface
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes

Figure 754: Data Format Structure for 3PR, G2, precipRateNearSurface

rainRateNearSurface
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes

Figure 755: Data Format Structure for 3PR, G2, rainRateNearSurface

snowRateNearSurface
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes

Figure 756: Data Format Structure for 3PR, G2, snowRateNearSurface

mixedPhRateNearSurface
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes

Figure 757: Data Format Structure for 3PR, G2, mixedPhRateNearSurface
precipWaterIntegrated
- count: 4 bytes, Array: ltH x lnH x chn x rt
- mean: 4 bytes, Array: ltH x lnH x chn x rt
- stdev: 4 bytes, Array: ltH x lnH x chn x rt

Figure 758: Data Format Structure for 3PR, G2, precipWaterIntegrated

precipIceIntegrated
- count: 4 bytes, Array: ltH x lnH x chn x rt
- mean: 4 bytes, Array: ltH x lnH x chn x rt
- stdev: 4 bytes, Array: ltH x lnH x chn x rt

Figure 759: Data Format Structure for 3PR, G2, precipIceIntegrated

precipRateAve24
- count: 4 bytes, Array: ltH x lnH x chn x rt
- mean: 4 bytes, Array: ltH x lnH x chn x rt
- stdev: 4 bytes, Array: ltH x lnH x chn x rt

Figure 760: Data Format Structure for 3PR, G2, precipRateAve24
Figure 761: Data Format Structure for 3PR, G2, zFactorCorrected

```
zFactorCorrected
  count 4 bytes
  mean 4 bytes
  stdev 4 bytes
```

Array: ltH x lnH x inst x hgt x rt

Figure 762: Data Format Structure for 3PR, G2, zFactorCorrectedESurface

```
zFactorCorrectedESurface
  count 4 bytes
  mean 4 bytes
  stdev 4 bytes
```

Array: ltH x lnH x inst x rt

Figure 763: Data Format Structure for 3PR, G2, zFactorCorrectedNearSurface

```
zFactorCorrectedNearSurface
  count 4 bytes
  mean 4 bytes
  stdev 4 bytes
```

Array: ltH x lnH x inst x rt

Figure 764: Data Format Structure for 3PR, G2, zFactorMeasuredNearSurface

```
zFactorMeasuredNearSurface
  count 4 bytes
  mean 4 bytes
  stdev 4 bytes
```

Array: ltH x lnH x inst x rt

Figure 765: Data Format Structure for 3PR, G2, zFactorCorrectedDPR

```
zFactorCorrectedDPR
  count 4 bytes
  mean 4 bytes
  stdev 4 bytes
```

Array: ltH x lnH x inst x hgt x rt

Figure 766: Data Format Structure for 3PR, G2, zFactorCorrectedESurfaceDPR

```
zFactorCorrectedESurfaceDPR
  count 4 bytes
  mean 4 bytes
  stdev 4 bytes
```

Array: ltH x lnH x inst x rt

Figure 767: Data Format Structure for 3PR, G2, zFactorCorrectedNearSurfaceDPR

```
zFactorCorrectedNearSurfaceDPR
  count 4 bytes
  mean 4 bytes
  stdev 4 bytes
```

Array: ltH x lnH x inst x rt
5 STANDARD GPM PRODUCTS

Figure 768: Data Format Structure for 3PR, G2, zFactorMeasured

- **zFactorMeasured**
  - count: 4 bytes
  - mean: 4 bytes
  - stdev: 4 bytes

Array: ltH x lnH x inst x hgt x rt

Figure 769: Data Format Structure for 3PR, G2, dm

- **dm**
  - count: 4 bytes
  - mean: 4 bytes
  - stdev: 4 bytes

Array: ltH x lnH x hgt x rt

Figure 770: Data Format Structure for 3PR, G2, dBNw

- **dBNw**
  - count: 4 bytes
  - mean: 4 bytes
  - stdev: 4 bytes

Array: ltH x lnH x hgt x rt
### 5.54 3PR - PR Full Product

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>epsilonDPR</td>
<td>count</td>
<td>4 bytes Array: ltH x lnH x inst x hgt x rt</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>4 bytes Array: ltH x lnH x inst x hgt x rt</td>
</tr>
<tr>
<td></td>
<td>stdev</td>
<td>4 bytes Array: ltH x lnH x inst x hgt x rt</td>
</tr>
<tr>
<td>epsilon</td>
<td>count</td>
<td>4 bytes Array: ltH x lnH x inst x rt</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>4 bytes Array: ltH x lnH x inst x rt</td>
</tr>
<tr>
<td></td>
<td>stdev</td>
<td>4 bytes Array: ltH x lnH x inst x rt</td>
</tr>
<tr>
<td>zeta</td>
<td>count</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>stdev</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td>piaHB</td>
<td>count</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>stdev</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td>piaHybrid</td>
<td>count</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>stdev</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td>piaHybridDPR</td>
<td>count</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>stdev</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td>piaSRT</td>
<td>count</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
<tr>
<td></td>
<td>stdev</td>
<td>4 bytes Array: ltH x lnH x inst x ang x rt</td>
</tr>
</tbody>
</table>

**Figure 771**: Data Format Structure for 3PR, G2, epsilonDPR

**Figure 772**: Data Format Structure for 3PR, G2, epsilon

**Figure 773**: Data Format Structure for 3PR, G2, zeta

**Figure 774**: Data Format Structure for 3PR, G2, piaHB

**Figure 775**: Data Format Structure for 3PR, G2, piaHybrid

**Figure 776**: Data Format Structure for 3PR, G2, piaHybridDPR

**Figure 777**: Data Format Structure for 3PR, G2, piaSRT
Figure 778: Data Format Structure for 3PR, G2, piaSRtdpr

Figure 779: Data Format Structure for 3PR, G2, piaFinal

Figure 780: Data Format Structure for 3PR, G2, piaFinalDPR
FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

Grids (Group)
G1 (Grid)

G1_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipRate (Group in G1)
Conditional Precipitation Rate.

count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

Figure 786: Data Format Structure for 3PR, G2, DFRNearSurface
hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin): Histogram. Special values are defined as:
-9999 Missing value

rainRate (Group in G1)
Conditional liquid water Rain Rate.

count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

snowRate (Group in G1)
Conditional Snowfall Rate.

count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

flagHeavyIcePrecip (Group in G1)
Counts of the occurrence of flagHeavyIcePrecip. Mean and std. dev. are set to missing.
The histogram contains counts of the integer flag values, with bins from 1 to 30.
count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

mixedPhRate (Group in G1)
Conditional Mixed Phase Precipitation Rate.

count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipRateESurface (Group in G1)
Conditional Estimated Surface Precipitation Rate.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value
**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipRateESurface2** (Group in G1)
Alternate Conditional Estimated Surface Precipitation Rate.

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipRateNearSurface** (Group in G1)
Conditional Precipitation Rate at Near Surface Level.

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value
**rainRateNearSurface** (Group in G1)
Unconditional liquid Rain Rate at Near Surface Level.

- **count** (4-byte integer, array size: ltL x lnL x chn x rt x st): Count. Special values are defined as:
  -9999 Missing value

- **mean** (4-byte float, array size: ltL x lnL x chn x rt x st): mean. Special values are defined as:
  -9999.9 Missing value

- **stdev** (4-byte float, array size: ltL x lnL x chn x rt x st): Standard deviation. Special values are defined as:
  -9999.9 Missing value

- **hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin): Histogram. Special values are defined as:
  -9999 Missing value

**snowRateNearSurface** (Group in G1)
Conditional Snow Rate at Near Surface Level.

- **count** (4-byte integer, array size: ltL x lnL x chn x rt x st): Count. Special values are defined as:
  -9999 Missing value

- **mean** (4-byte float, array size: ltL x lnL x chn x rt x st): mean. Special values are defined as:
  -9999.9 Missing value

- **stdev** (4-byte float, array size: ltL x lnL x chn x rt x st): Standard deviation. Special values are defined as:
  -9999.9 Missing value

- **hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin): Histogram. Special values are defined as:
  -9999 Missing value

**mixedPhRateNearSurface** (Group in G1)
Conditional Mixed Phase Precipitation Rate at Near Surface Level.

- **count** (4-byte integer, array size: ltL x lnL x chn x rt x st): Count. Special values are defined as:
  -9999 Missing value
**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
Mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**precipWaterIntegrated** (Group in G1)
Integrated Precipitable Water ($g/m^2$).

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
Mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**precipIceIntegrated** (Group in G1)
Integrated Precipitable Ice

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
Mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value
hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin): Histogram. Special values are defined as:
-9999  Missing value

**precipRateAve24** (Group in G1)
Average Precipitation Rate in 24hrs.

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st): Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st): mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st): Standard deviation. Special values are defined as:
-9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin): Histogram. Special values are defined as:
-9999  Missing value

**zFactorCorrected** (Group in G1)
Corrected Reflectivity

**count** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st): Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st): mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st): Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin): Histogram. Special values are defined as:
-9999  Missing value

**zFactorCorrectedESurface** (Group in G1)
Corrected Reflectivity at the Estimated Surface
5.54  3PR - PR Full Product

**count** (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

---

**zFactorCorrectedNearSurface** (Group in G1)
Corrected Reflectivity at the Near Surface Level.

**count** (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

---

**zFactorMeasuredNearSurface** (Group in G1)
Measured Reflectivity at the Near Surface Level.

**count** (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9  Missing value
**stdev** (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

---

**zFactorCorrectedDPR** (Group in G1)
Corrected Reflectivity from DPR

**count** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

---

**zFactorCorrectedESurfaceDPR** (Group in G1)
Corrected Reflectivity from DPR at Estimated Surface.

**count** (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value
**zFactorCorrectedNearSurfaceDPR** (Group in G1)
Corrected Reflectivity from DPR at the Near Surface Level.

**count** (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**zFactorMeasured** (Group in G1)
Measured Reflectivity

**count** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**dm** (Group in G1)

**count** (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value
mean (4-byte float, array size: ltL x lnL x hgt x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

dBNw (Group in G1)

count (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltL x lnL x hgt x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

epsilonDPR (Group in G1)

count (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value
hist (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

epsilon (Group in G1)

count (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

zeta (Group in G1)
Integral of $0.2 \ln(10) \alpha Zm^{beta}$ over the slant range path where $\alpha$ and $Zm$ are functions of range.

count (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

piaHB (Group in G1)
Hitchfield-Bordan Path Integrated Attenuation for the slant range path.
count (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

piaHybrid (Group in G1)
Weighted Hybrid PIA between the HB solution and the SRT PIA.

count (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

piaHybridDPR (Group in G1)
Weighted Hybrid PIA between the HB solution and the SRT PIA for DPR.

count (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
-9999.9 Missing value
\textbf{stdev} (4-byte float, array size: ltL x lnL x inst x ang x rt x st): Standard deviation. Special values are defined as:
-9999.9 Missing value

\textbf{hist} (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin): Histogram. Special values are defined as:
-9999 Missing value

\textbf{piaSRT} (Group in G1)
Path Integrated Attenuation from SRT.
\textbf{count} (4-byte integer, array size: ltL x lnL x inst x ang x rt x st): Count. Special values are defined as:
-9999 Missing value

\textbf{mean} (4-byte float, array size: ltL x lnL x inst x ang x rt x st): mean. Special values are defined as:
-9999.9 Missing value

\textbf{stdev} (4-byte float, array size: ltL x lnL x inst x ang x rt x st): Standard deviation. Special values are defined as:
-9999.9 Missing value

\textbf{hist} (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin): Histogram. Special values are defined as:
-9999 Missing value

\textbf{piaSRTDpr} (Group in G1)
Path Integrated Attenuation from SRT DPR
\textbf{count} (4-byte integer, array size: ltL x lnL x inst x ang x rt x st): Count. Special values are defined as:
-9999 Missing value

\textbf{mean} (4-byte float, array size: ltL x lnL x inst x ang x rt x st): mean. Special values are defined as:
-9999.9 Missing value

\textbf{stdev} (4-byte float, array size: ltL x lnL x inst x ang x rt x st): Standard deviation. Special values are defined as:
-9999.9 Missing value

\textbf{hist} (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin): Histogram. Special values are defined as:
-9999 Missing value
piaFinal (Group in G1)
Final Path Integrated Attenuation

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
- 9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
- 9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
- 9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
- 9999  Missing value

piaFinalDPR (Group in G1)
Final Path Integrated Attenuation from DPR

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
- 9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
- 9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
- 9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
- 9999  Missing value

piaFinalSubset (Group in G1)
Final Path Integrated Attenuation Subset

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
- 9999  Missing value
**mean** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\): mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\): Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: \(ltL \times lnL \times inst \times ang \times rt \times st \times bin\): Histogram. Special values are defined as:
-9999 Missing value

**piaFinalDPRsubset** (Group in G1)
Final Path Integrated Attenuation from DPR Subset

**count** (4-byte integer, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\): Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\): mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\): Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: \(ltL \times lnL \times inst \times ang \times rt \times st \times bin\): Histogram. Special values are defined as:
-9999 Missing value

**heightBB** (Group in G1)
Height of Bright Band.

**count** (4-byte integer, array size: \(ltL \times lnL \times chn \times rt \times st\): Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: \(ltL \times lnL \times chn \times rt \times st\): mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: \(ltL \times lnL \times chn \times rt \times st\): Standard deviation. Special values are defined as:
-9999.9 Missing value
hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

heightBBnadir (Group in G1)
Height of Bright Band from Nadir.
count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999  Missing value
mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9  Missing value
stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value
hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

BBwidthNadir (Group in G1)
Width of Bright Band at Nadir

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999  Missing value
mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9  Missing value
stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value
hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

heightStormTop (Group in G1)
Storm Top Height
count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
  -9999  Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
  -9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
  -9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
  -9999  Missing value

BBwidth (Group in G1)
Bright Band Width

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
  -9999  Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
  -9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
  -9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
  -9999  Missing value

observationCounts (Group in G1)
Observation Counts

total (4-byte integer, array size: ltL x lnL x inst x st):
Total obs. Special values are defined as:
  -9999  Missing value

localTime (4-byte integer, array size: ltL x lnL x inst x tim x st):
obs time. Special values are defined as:
  -9999  Missing value
**pia** (4-byte integer, array size: ltL x lnL x inst x ang x st):
obs PIA. Special values are defined as:
- 9999 Missing value

**shallowRain** (4-byte integer, array size: ltL x lnL x inst x st):
obs time. Special values are defined as:
- 9999 Missing value

**precipRateLocalTime** (Group in G1)
Precipitation Rate by Local Time

**count** (4-byte integer, array size: ltL x lnL x chn x tim x st):
Count. Special values are defined as:
- 9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x tim x st):
mean. Special values are defined as:
- 9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x tim x st):
Standard deviation. Special values are defined as:
- 9999.9 Missing value

**DFRmNearSurface** (Group in G1)
DFRm at the Near Surface level

**count** (4-byte integer, array size: ltL x lnL x rt x st):
Count. Special values are defined as:
- 9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x rt x st):
mean. Special values are defined as:
- 9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x rt x st):
Standard deviation. Special values are defined as:
- 9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x rt x st x bin):
Histogram. Special values are defined as:
- 9999 Missing value

**DFRNearSurface** (Group in G1)
DFR at the Near Surface level
count (4-byte integer, array size: ltL x lnL x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipRateNearSurfaceUnconditional (4-byte float, array size: ltL x lnL x chn):
Rain, not conditioned on rain. Special values are defined as:
-9999.9 Missing value

precipProbabilityNearSurface (4-byte float, array size: ltL x lnL x chn):
Probability of rain. Special values are defined as:
-9999.9 Missing value

G2 (Grid)

G2_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipRate (Group in G2)
Conditional Precipitation Rate

count (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**rainRate** (Group in G2)
Conditional Liquid Rain Rate

**count** (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**snowRate** (Group in G2)
Conditional Snow Rate

**count** (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**flagHeavyIcePrecip** (Group in G2)
Counts of the occurrence of flagHeavyIcePrecip. Mean and std. dev. are set to missing.
The histogram contains counts of the integer flag values, with bins from 1 to 30.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value
mixedPhRate (Group in G2)
Conditional Precipitation Rate of Mixed Phase

count (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

precipRateESurface (Group in G2)
Conditional Estimated Precipitation Rate at the Surface

count (4-byte integer, array size: ltH x lnH x chn x x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

precipRateESurface2 (Group in G2)
Alternate Conditional Estimated Precipitation Rate at the Surface

count (4-byte integer, array size: ltH x lnH x chn x x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
precipRateNearSurface (Group in G2)
Conditional Precipitation Rate at the Near Surface Level.

count (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

rainRateNearSurface (Group in G2)
Conditional Liquid Rain Rate at the Near Surface Level.

count (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

snowRateNearSurface (Group in G2)
Conditional Snow Rate at the Near Surface Level.

count (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**mixedPhRateNearSurface** (Group in G2)
Conditional Precipitation Rate of Mixed Phase at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**precipWaterIntegrated** (Group in G2)
Integrated Precipitable Water \( (g/m^2) \).

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**precipIceIntegrated** (Group in G2)
Integrated Precipitable Ice

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
precipRateAve24 (Group in G2)
Conditional Precipitation Rate Averaged for 24hrs.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

zFactorCorrected (Group in G2)
Corrected Reflectivity.

**count** (4-byte integer, array size: ltH x lnH x inst x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

zFactorCorrectedESurface (Group in G2)
Corrected Reflectivity Estimate at the Surface

**count** (4-byte integer, array size: ltH x lnH x inst x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**zFactorCorrectedNearSurface** (Group in G2)
Corrected Reflectivity at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x inst x rt): 
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x rt): 
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x rt): 
Standard deviation. Special values are defined as:
-9999.9 Missing value

**zFactorMeasuredNearSurface** (Group in G2)
Measured Reflectivity at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x inst x rt): 
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x rt): 
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x rt): 
Standard deviation. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedDPR** (Group in G2)
Corrected Reflectivity from DPR

**count** (4-byte integer, array size: ltH x lnH x inst x hgt x rt): 
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x hgt x rt): 
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x hgt x rt): 
Standard deviation. Special values are defined as:
-9999.9 Missing value
**zFactorCorrectedESurfaceDPR** (Group in G2)
Estimated Corrected Reflectivity at the Surface

**count** (4-byte integer, array size: ltH x lnH x inst x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurfaceDPR** (Group in G2)
Corrected Reflectivity at the Near Surface Level for DPR

**count** (4-byte integer, array size: ltH x lnH x inst x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**zFactorMeasured** (Group in G2)
Corrected Reflectivity

**count** (4-byte integer, array size: ltH x lnH x inst x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**dm** (Group in G2)
Mean Mass-Weighted Drop Diameter

**count** (4-byte integer, array size: ltH x lnH x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**dBNw** (Group in G2)
Normalized Drop Concentration Parameter

**count** (4-byte integer, array size: ltH x lnH x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**epsilonDPR** (Group in G2)

**count** (4-byte integer, array size: ltH x lnH x inst x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**epsilon** (Group in G2)

- **count** (4-byte integer, array size: `ltH x lnH x inst x rt`):
  Count. Special values are defined as:
  - `-9999` Missing value

- **mean** (4-byte float, array size: `ltH x lnH x inst x rt`):
  mean. Special values are defined as:
  - `-9999.9` Missing value

- **stdev** (4-byte float, array size: `ltH x lnH x inst x rt`):
  Standard deviation. Special values are defined as:
  - `-9999.9` Missing value

---

**zeta** (Group in G2)

Integral of \(0.2 \cdot \ln(10) \cdot \alpha \cdot Zm^{\beta} \) over the slant range path where \(\alpha\) and \(Zm\) are functions of range.

- **count** (4-byte integer, array size: `ltH x lnH x inst x ang x rt`):
  Count. Special values are defined as:
  - `-9999` Missing value

- **mean** (4-byte float, array size: `ltH x lnH x inst x ang x rt`):
  mean. Special values are defined as:
  - `-9999.9` Missing value

- **stdev** (4-byte float, array size: `ltH x lnH x inst x ang x rt`):
  Standard deviation. Special values are defined as:
  - `-9999.9` Missing value

---

**piaHB** (Group in G2)

Hitchfield-Bordan Path Integrated Attenuation for the slant range path.

- **count** (4-byte integer, array size: `ltH x lnH x inst x ang x rt`):
  Count. Special values are defined as:
  - `-9999` Missing value

- **mean** (4-byte float, array size: `ltH x lnH x inst x ang x rt`):
  mean. Special values are defined as:
  - `-9999.9` Missing value

- **stdev** (4-byte float, array size: `ltH x lnH x inst x ang x rt`):
  Standard deviation. Special values are defined as:
  - `-9999.9` Missing value
piaHybrid (Group in G2)
Weighted Hybrid PIA between the HB solution and the SRT PIA.

count (4-byte integer, array size: ltH x lnH x inst x ang x rt):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltH x lnH x inst x ang x rt):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltH x lnH x inst x ang x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

piaHybridDPR (Group in G2)
Weighted Hybrid PIA between the HB solution and the SRT PIA for DPR.

count (4-byte integer, array size: ltH x lnH x inst x ang x rt):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltH x lnH x inst x ang x rt):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltH x lnH x inst x ang x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

piaSRT (Group in G2)
Path Integrated Attenuation from SRT.

count (4-byte integer, array size: ltH x lnH x inst x ang x rt):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltH x lnH x inst x ang x rt):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltH x lnH x inst x ang x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value
**piaSRTdpr** (Group in G2)
Path Integrated Attenuation from SRT for DPR.

**count** (4-byte integer, array size: ltH x lnH x inst x ang x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x ang x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**piaFinal** (Group in G2)
Final Path Integrated Attenuation Estimate.

**count** (4-byte integer, array size: ltH x lnH x inst x ang x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x ang x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**piaFinalDPR** (Group in G2)
Final Path Integrated Attenuation Estimate for DPR.

**count** (4-byte integer, array size: ltH x lnH x inst x ang x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x ang x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**heightBB** (Group in G2)
Height Of the Bright Band.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
- 9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
- 9999.9 Missing value

**stddev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
- 9999.9 Missing value

**heightStormTop** (Group in G2)
Height of the Storm Top.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
- 9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
- 9999.9 Missing value

**stddev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
- 9999.9 Missing value

**BBwidth** (Group in G2)
Bright Band Width

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
- 9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
- 9999.9 Missing value

**stddev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
- 9999.9 Missing value
observationCounts (Group in G2)
Observation Counts.

total (4-byte integer, array size: ltH x lnH x inst):
Total obs. Special values are defined as:
-9999 Missing value

pia (4-byte integer, array size: ltH x lnH x inst x ang):
obs PIA. Special values are defined as:
-9999 Missing value

shallowRain (4-byte integer, array size: ltH x lnH x inst):
obs time. Special values are defined as:
-9999 Missing value

DFRmNearSurface (Group in G2)
DFRm at the Near Surface level

count (4-byte integer, array size: ltH x lnH x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

DFRNearSurface (Group in G2)
DFR at the Near Surface level

count (4-byte integer, array size: ltH x lnH x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
precipRateNearSurfaceUnconditional (4-byte float, array size: ltH x lnH x chn): Rain, not conditioned on rain. Special values are defined as:
   -9999.9 Missing value

precipProbabilityNearSurface (4-byte float, array size: ltH x lnH x chn): Probability of rain. Special values are defined as:
   -9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_3PR_H_
define _TK_3PR_H_

#ifndef _L3PR_G2_DFRNEARSURFACE_
define _L3PR_G2_DFRNEARSURFACE_
typedef struct {
    int count[3][1440][536];
    float mean[3][1440][536];
    float stdev[3][1440][536];
} L3PR_G2_DFRNEARSURFACE;

#define _L3PR_G2_DFRNEARSURFACE_

#ifndef _L3PR_G2_DFRMNEARSURFACE_
define _L3PR_G2_DFRMNEARSURFACE_
typedef struct {
    int count[3][1440][536];
    float mean[3][1440][536];
    float stdev[3][1440][536];
} L3PR_G2_DFRMNEARSURFACE;

#define _L3PR_G2_DFRMNEARSURFACE_

#ifndef _L3PR_G2_OBSERVATIONCOUNTS_
define _L3PR_G2_OBSERVATIONCOUNTS_
typedef struct {
    int total[4][1440][536];
    int pia[7][4][1440][536];
    int shallowRain[4][1440][536];
} L3PR_G2_OBSERVATIONCOUNTS;

#define _L3PR_G2_OBSERVATIONCOUNTS_
```

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_BBWIDTH;

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_HEIGHTSTORMTOP;

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_HEIGHTBB;

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3PR_G2_PIAFINALDPR;
#ifndef _L3PR_G2_PIAFINAL_
define _L3PR_G2_PIAFINAL_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3PR_G2_PIAFINAL;
#endif

#ifndef _L3PR_G2_PIASRTDPR_
define _L3PR_G2_PIASRTDPR_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3PR_G2_PIASRTDPR;
#endif

#ifndef _L3PR_G2_PIASRT_
define _L3PR_G2_PIASRT_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3PR_G2_PIASRT;
#endif

#ifndef _L3PR_G2_PIAHYBRIDDPR_
define _L3PR_G2_PIAHYBRIDDPR_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3PR_G2_PIAHYBRIDDPR;
#endif
} L3PR_G2_PIAHYBRID;
#endif

#ifndef _L3PR_G2_PIAHYBRID_
#define _L3PR_G2_PIAHYBRID_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3PR_G2_PIAHYBRID;
#endif

#ifndef _L3PR_G2_PIAHB_
#define _L3PR_G2_PIAHB_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3PR_G2_PIAHB;
#endif

#ifndef _L3PR_G2_ZETA_
#define _L3PR_G2_ZETA_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3PR_G2_ZETA;
#endif

#ifndef _L3PR_G2_EPSILON_
#define _L3PR_G2_EPSILON_

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
} L3PR_G2_EPSILON;
float stdev[3][4][1440][536];
} L3PR_G2_EPSILON;

#endif

#ifndef _L3PR_G2_EPSILON_DPR_
#define _L3PR_G2_EPSILON_DPR_

typedef struct {
    int count[3][5][4][1440][536];
    float mean[3][5][4][1440][536];
    float stdev[3][5][4][1440][536];
} L3PR_G2_EPSILON_DPR;

#endif

#ifndef _L3PR_G2_DBNW_
#define _L3PR_G2_DBNW_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_DBNW;

#endif

#ifndef _L3PR_G2_DM_
#define _L3PR_G2_DM_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_DM;

#endif

#ifndef _L3PR_G2_ZFACTOR_MEASURED_
#define _L3PR_G2_ZFACTOR_MEASURED_

typedef struct {
    int count[3][5][4][1440][536];

}
float mean[3][5][4][1440][536];
float stdev[3][5][4][1440][536];
} L3PR_G2_ZFACTORMEASURED;
#endif

#ifndef _L3PR_G2_ZFACTORCORRECTEDNEARSURFACEDPR_
#define _L3PR_G2_ZFACTORCORRECTEDNEARSURFACEDPR_

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3PR_G2_ZFACTORCORRECTEDNEARSURFACEDPR;
#endif

#ifndef _L3PR_G2_ZFACTORCORRECTEDSURFACEDPR_
#define _L3PR_G2_ZFACTORCORRECTEDSURFACEDPR_

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3PR_G2_ZFACTORCORRECTEDSURFACEDPR;
#endif

#ifndef _L3PR_G2_ZFACTORCORRECTEDDPR_
#define _L3PR_G2_ZFACTORCORRECTEDDPR_

typedef struct {
    int count[3][5][4][1440][536];
    float mean[3][5][4][1440][536];
    float stdev[3][5][4][1440][536];
} L3PR_G2_ZFACTORCORRECTEDDPR;
#endif

#ifndef _L3PR_G2_ZFACTORMEASUREDNEARSURFACE_
#define _L3PR_G2_ZFACTORMEASUREDNEARSURFACE_

typedef struct {
typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3PR_G2_ZFACTORMEASUREDNEARSURFACE;

#endif

#ifndef _L3PR_G2_ZFACTORCORRECTEDNEARSURFACE_
define _L3PR_G2_ZFACTORCORRECTEDNEARSURFACE_

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3PR_G2_ZFACTORCORRECTEDNEARSURFACE;

#endif

#ifndef _L3PR_G2_ZFACTORCORRECTEDESURFACE_
define _L3PR_G2_ZFACTORCORRECTEDESURFACE_

typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3PR_G2_ZFACTORCORRECTEDESURFACE;

#endif

#ifndef _L3PR_G2_ZFACTORCORRECTED_
define _L3PR_G2_ZFACTORCORRECTED_

typedef struct {
    int count[3][5][4][1440][536];
    float mean[3][5][4][1440][536];
    float stdev[3][5][4][1440][536];
} L3PR_G2_ZFACTORCORRECTED;

#endif

#ifndef _L3PR_G2_PRECIPRATEAVE24_
define _L3PR_G2_PRECIPRATEAVE24_


typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_PRECIPRATEAVE24;

#endif

#ifndef _L3PR_G2_PRECIPICEINTEGRATED_
#define _L3PR_G2_PRECIPICEINTEGRATED_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_PRECIPICEINTEGRATED;
#endif

#ifndef _L3PR_G2_PRECIPWATERINTEGRATED_
#define _L3PR_G2_PRECIPWATERINTEGRATED_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_PRECIPWATERINTEGRATED;
#endif

#ifndef _L3PR_G2_MIXEDPHRATENEARSURFACE_
#define _L3PR_G2_MIXEDPHRATENEARSURFACE_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_MIXEDPHRATENEARSURFACE;
#endif

#ifndef _L3PR_G2_SNOWRATENEARSURFACE_
#define _L3PR_G2_SNOWRATENEARSURFACE_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_SNOWRATENEARSURFACE;
#endif
typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_SNOWRATENEARSURFACE;

#endif

#ifndef _L3PR_G2_RAINRATENEARSURFACE_
#define _L3PR_G2_RAINRATENEARSURFACE_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_RAINRATENEARSURFACE;

#endif

#ifndef _L3PR_G2_PRECIPRATENEARSURFACE_
#define _L3PR_G2_PRECIPRATENEARSURFACE_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_PRECIPRATENEARSURFACE;

#endif

#ifndef _L3PR_G2_PRECIPRATEESURFACE2_
#define _L3PR_G2_PRECIPRATEESURFACE2_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_PRECIPRATEESURFACE2;

#endif
typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_PRECIPRATEESURFACE;

#endif

#ifndef _L3PR_G2_MIXEDPHRATE_
#define _L3PR_G2_MIXEDPHRATE_

typedef struct {
    int count[3][5][5][1440][536];
    float mean[3][5][5][1440][536];
    float stdev[3][5][5][1440][536];
} L3PR_G2_MIXEDPHRATE;
#endif

#ifndef _L3PR_G2_FLAGHEAVYICEPRECIP_
#define _L3PR_G2_FLAGHEAVYICEPRECIP_

typedef struct {
    int count[3][5][1440][536];
    float mean[3][5][1440][536];
    float stdev[3][5][1440][536];
} L3PR_G2_FLAGHEAVYICEPRECIP;
#endif

#ifndef _L3PR_G2_SNOWRATE_
#define _L3PR_G2_SNOWRATE_

typedef struct {
    int count[3][5][5][1440][536];
    float mean[3][5][5][1440][536];
    float stdev[3][5][5][1440][536];
} L3PR_G2_SNOWRATE;
#endif
```c
#ifndef _L3PR_G2_RAINRATE_
define _L3PR_G2_RAINRATE_

typedef struct {
    int count[3][5][5][1440][536];
    float mean[3][5][5][1440][536];
    float stdev[3][5][5][1440][536];
} L3PR_G2_RAINRATE;
#endif

#ifndef _L3PR_G2_PRECIPRATE_
define _L3PR_G2_PRECIPRATE_

typedef struct {
    int count[3][5][5][1440][536];
    float mean[3][5][5][1440][536];
    float stdev[3][5][5][1440][536];
} L3PR_G2_PRECIPRATE;
#endif

#ifndef _L3PR_G2_
define _L3PR_G2_

typedef struct {
    L3PR_G2_PRECIPRATE precipRate;
    L3PR_G2_RAINRATE rainRate;
    L3PR_G2_SNOWRATE snowRate;
    L3PR_G2_FLAGHEAVYICEPRECIP flagHeavyIcePrecip;
    L3PR_G2_MIXEDPHRATE mixedPhRate;
    L3PR_G2_PRECIPRATEESURFACE precipRateESurface;
    L3PR_G2_PRECIPRATEESURFACE2 precipRateESurface2;
    L3PR_G2_PRECIPRATENEARSURFACE precipRateNearSurface;
    L3PR_G2_RAINRATENEARSURFACE rainRateNearSurface;
    L3PR_G2_SNOWRATENEARSURFACE snowRateNearSurface;
    L3PR_G2_MIXEDPHRATENEARSURFACE mixedPhRateNearSurface;
    L3PR_G2_PRECIPWATERINTEGRATED precipWaterIntegrated;
    L3PR_G2_PRECIPICEINTEGRATED precipIceIntegrated;
    L3PR_G2_PRECIPRATEAVE24 precipRateAve24;
    L3PR_G2_ZFACTORCORRECTED zFactorCorrected;
    L3PR_G2_ZFACTORCORRECTEDESURFACE zFactorCorrectedESurface;
    L3PR_G2_ZFACTORCORRECTEDNEARSURFACE zFactorCorrectedNearSurface;
```
typedef struct {
    int count[3][3][72][28];
    float mean[3][3][72][28];
    float stdev[3][3][72][28];
    int hist[30][3][3][72][28];
} L3PR_G1_DFRNEARSURFACE;

#ifdef _L3PR_G1_DFRMNEARSURFACE_
#define _L3PR_G1_DFRMNEARSURFACE_  

typedef struct {
    int count[3][3][72][28];
    float mean[3][3][72][28];
    float stdev[3][3][72][28];
    int hist[30][3][3][72][28];
} L3PR_G1_DFRMNEARSURFACE;

#endif

#ifndef _L3PR_G2_ZFACTORMEASUREDNEARSURFACE_
#define _L3PR_G2_ZFACTORMEASUREDNEARSURFACE_  

L3PR_G2_ZFACTORMEASUREDNEARSURFACE zFactorMeasuredNearSurface;
L3PR_G2_ZFACTORCORRECTEDDPR zFactorCorrectedDPR;
L3PR_G2_ZFACTORCORRECTEDESURFACEDPR zFactorCorrectedESurfaceDPR;
L3PR_G2_ZFACTORCORRECTEDNEARSURFACEDPR zFactorCorrectedNearSurfaceDPR;
L3PR_G2_ZFACTORMEASURED zFactorMeasured;
L3PR_G2_DM dm;
L3PR_G2_DBNW dBNw;
L3PR_G2_EPSILONDPR epsilonDPR;
L3PR_G2_EPSILON epsilon;
L3PR_G2_ZETA zeta;
L3PR_G2_PIAHB piaHB;
L3PR_G2_PIAHYBRID piaHybrid;
L3PR_G2_PIAHYBRIDDPR piaHybridDPR;
L3PR_G2_PIASRT piaSRT;
L3PR_G2_PIASRTDPR piaSRTdpr;
L3PR_G2_PIAFINAL piaFinal;
L3PR_G2_PIAFINALDPR piaFinalDPR;
L3PR_G2_HEIGHTBB heightBB;
L3PR_G2_HEIGHTSTORMTOP heightStormTop;
L3PR_G2_BBWIDTH BBwidth;
L3PR_G2_OBSERVATIONCOUNTS observationCounts;
L3PR_G2_DFRMNEARSURFACE DFRmNearSurface;
L3PR_G2_DFRNEARSURFACE DFRNearSurface;
float precipRateNearSurfaceUnconditional[5][1440][536];
float precipProbabilityNearSurface[5][1440][536];
} L3PR_G2;

#endif

#elifndef _L3PR_G1_DFRNEARSURFACE_
#define _L3PR_G1_DFRNEARSURFACE_  

L3PR_G1_DFRNEARSURFACE DFRNearSurface;
float precipRateNearSurfaceUnconditional[5][1440][536];
float precipProbabilityNearSurface[5][1440][536];
} L3PR_G1_DFRNEARSURFACE;

#endif

#elifndef _L3PR_G1_DFRMNEARSURFACE_
#define _L3PR_G1_DFRMNEARSURFACE_  

L3PR_G1_DFRMNEARSURFACE DFRmNearSurface;
float precipRateNearSurfaceUnconditional[5][1440][536];
float precipProbabilityNearSurface[5][1440][536];
} L3PR_G1_DFRMNEARSURFACE;

#endif
typedef struct {
    int count[3][3][72][28];
    float mean[3][3][72][28];
    float stdev[3][3][72][28];
    int hist[30][3][3][72][28];
} L3PR_G1_DFRMNEARSURFACE;

#endif

#ifndef _L3PR_G1_PRECIPRATELOCALTIME_
#define _L3PR_G1_PRECIPRATELOCALTIME_

typedef struct {
    int count[3][24][5][72][28];
    float mean[3][24][5][72][28];
    float stdev[3][24][5][72][28];
} L3PR_G1_PRECIPRATELOCALTIME;

#endif

#ifndef _L3PR_G1_OBSERVATIONCOUNTS_
#define _L3PR_G1_OBSERVATIONCOUNTS_

typedef struct {
    int total[3][4][72][28];
    int localTime[3][24][4][72][28];
    int pia[3][7][4][72][28];
    int shallowRain[3][4][72][28];
} L3PR_G1_OBSERVATIONCOUNTS;

#endif

#ifndef _L3PR_G1_BBWIDTH_
#define _L3PR_G1_BBWIDTH_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_BBWIDTH;
#ifndef _L3PR_G1_HEIGHTSTORMTOP_
#define _L3PR_G1_HEIGHTSTORMTOP_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_HEIGHTSTORMTOP;
#endif

#ifndef _L3PR_G1_BBWIDTHNADIR_
#define _L3PR_G1_BBWIDTHNADIR_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_BBWIDTHNADIR;
#endif

#ifndef _L3PR_G1_HEIGHTBBNADIR_
#define _L3PR_G1_HEIGHTBBNADIR_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_HEIGHTBBNADIR;
#endif

#ifndef _L3PR_G1_HEIGHTBB_
#define _L3PR_G1_HEIGHTBB_

typedef struct {
    int count[3][3][5][72][28];
}
float mean[3][3][5][72][28];
float stdev[3][3][5][72][28];
int hist[30][3][3][5][72][28];
} L3PR_G1_HEIGHTBB;

#endif

#ifndef _L3PR_G1_PIAFINALDPRSUBSET_
define _L3PR_G1_PIAFINALDPRSUBSET_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3PR_G1_PIAFINALDPRSUBSET;
#endif

#ifndef _L3PR_G1_PIAFINALSUBSET_
define _L3PR_G1_PIAFINALSUBSET_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3PR_G1_PIAFINALSUBSET;
#endif

#ifndef _L3PR_G1_PIAFINALDPR_
define _L3PR_G1_PIAFINALDPR_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3PR_G1_PIAFINALDPR;
#endif
#ifndef _L3PR_G1_PIAFINAL_
define _L3PR_G1_PIAFINAL_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3PR_G1_PIAFINAL;
#endif

#ifndef _L3PR_G1_PIASRTDPR_
define _L3PR_G1_PIASRTDPR_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3PR_G1_PIASRTDPR;
#endif

#ifndef _L3PR_G1_PIASRT_
define _L3PR_G1_PIASRT_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3PR_G1_PIASRT;
#endif

#ifndef _L3PR_G1_PIAHYBRIDDPR_
define _L3PR_G1_PIAHYBRIDDPR_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
} L3PR_G1_PIAHYBRIDDPR;
#endif
int hist[30][3][3][7][4][72][28];
}
L3PR_G1_PIAHYBRID;
#endif
#endif
#define _L3PR_G1_PIAHYBRID_

typedef struct {
  int count[3][3][7][4][72][28];
  float mean[3][3][7][4][72][28];
  float stdev[3][3][7][4][72][28];
  int hist[30][3][3][7][4][72][28];
} L3PR_G1_PIAHYBRID;
#endif
#endif
#define _L3PR_G1_PIAHB_

typedef struct {
  int count[3][3][7][4][72][28];
  float mean[3][3][7][4][72][28];
  float stdev[3][3][7][4][72][28];
  int hist[30][3][3][7][4][72][28];
} L3PR_G1_PIAHB;
#endif
#endif
#define _L3PR_G1_ZETA_

typedef struct {
  int count[3][3][7][4][72][28];
  float mean[3][3][7][4][72][28];
  float stdev[3][3][7][4][72][28];
  int hist[30][3][3][7][4][72][28];
} L3PR_G1_ZETA;
#endif
#endif
#define _L3PR_G1_EPSILON_

typedef struct {
  int count[3][3][7][4][72][28];
  float mean[3][3][7][4][72][28];
  float stdev[3][3][7][4][72][28];
  int hist[30][3][3][7][4][72][28];
} L3PR_G1_EPSILON;
typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3PR_G1_EPSILON;

#endif

#ifndef _L3PR_G1_EPSILON_DPR_
#define _L3PR_G1_EPSILON_DPR_

typedef struct {
    int count[3][3][5][4][72][28];
    float mean[3][3][5][4][72][28];
    float stdev[3][3][5][4][72][28];
    int hist[30][3][3][5][4][72][28];
} L3PR_G1_EPSILON_DPR;

#endif

#ifndef _L3PR_G1_DBNW_
#define _L3PR_G1_DBNW_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_DBNW;

#endif

#ifndef _L3PR_G1_DM_
#define _L3PR_G1_DM_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_DM;
typedef struct {
    int count[3][3][5][4][72][28];
    float mean[3][3][5][4][72][28];
    float stdev[3][3][5][4][72][28];
    int hist[30][3][3][5][4][72][28];
} L3PR_G1_ZFACTORMEASURED;

#endif

#ifndef _L3PR_G1_ZFACTORCORRECTEDNEARSURFACEDPR_
#define _L3PR_G1_ZFACTORCORRECTEDNEARSURFACEDPR_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3PR_G1_ZFACTORCORRECTEDNEARSURFACEDPR;

#endif

#ifndef _L3PR_G1_ZFACTORCORRECTEDSURFACEDPR_
#define _L3PR_G1_ZFACTORCORRECTEDSURFACEDPR_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3PR_G1_ZFACTORCORRECTEDSURFACEDPR;

#endif

#ifndef _L3PR_G1_ZFACTORCORRECTEDDPR_
#define _L3PR_G1_ZFACTORCORRECTEDDPR_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3PR_G1_ZFACTORCORRECTEDDPR;
int count[3][3][5][4][72][28];
float mean[3][3][5][4][72][28];
float stdev[3][3][5][4][72][28];
int hist[30][3][3][5][4][72][28];
} L3PR_G1_ZFACTORCORRECTEDDPR;

#endif

#ifndef _L3PR_G1_ZFACTORMEASUREDNEARSURFACE_
#define _L3PR_G1_ZFACTORMEASUREDNEARSURFACE_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3PR_G1_ZFACTORMEASUREDNEARSURFACE;

#endif

#ifndef _L3PR_G1_ZFACTORCORRECTEDNEARSURFACE_
#define _L3PR_G1_ZFACTORCORRECTEDNEARSURFACE_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3PR_G1_ZFACTORCORRECTEDNEARSURFACE;

#endif

#ifndef _L3PR_G1_ZFACTORCORRECTEDSURFACE_
#define _L3PR_G1_ZFACTORCORRECTEDSURFACE_

typedef struct {
    int count[3][3][4][72][28];
    float mean[3][3][4][72][28];
    float stdev[3][3][4][72][28];
    int hist[30][3][3][4][72][28];
} L3PR_G1_ZFACTORCORRECTEDSURFACE;

#endif
typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_ZFACTORCORRECTED;

#endif

#ifndef _L3PR_G1_PRECIPRATEAVE24_
#define _L3PR_G1_PRECIPRATEAVE24_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_PRECIPRATEAVE24;

#endif

#ifndef _L3PR_G1_PRECIPICEINTEGRATED_
#define _L3PR_G1_PRECIPICEINTEGRATED_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_PRECIPICEINTEGRATED;

#endif

#ifndef _L3PR_G1_PRECIPWATERINTEGRATED_
#define _L3PR_G1_PRECIPWATERINTEGRATED_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];

float stddev[3][3][5][72][28];
int hist[30][3][3][5][72][28];
} L3PR_G1_PRECIPWATERINTEGRATED;

#endif

#ifndef _L3PR_G1_MIXEDPHRATENEARSURFACE_
#define _L3PR_G1_MIXEDPHRATENEARSURFACE_

typedef struct {
  int count[3][3][5][72][28];
  float mean[3][3][5][72][28];
  float stddev[3][3][5][72][28];
  int hist[30][3][3][5][72][28];
} L3PR_G1_MIXEDPHRATENEARSURFACE;

#endif

#ifndef _L3PR_G1_SNOWRATENEARSURFACE_
#define _L3PR_G1_SNOWRATENEARSURFACE_

typedef struct {
  int count[3][3][5][72][28];
  float mean[3][3][5][72][28];
  float stddev[3][3][5][72][28];
  int hist[30][3][3][5][72][28];
} L3PR_G1_SNOWRATENEARSURFACE;

#endif

#ifndef _L3PR_G1_RAINRATENEARSURFACE_
#define _L3PR_G1_RAINRATENEARSURFACE_

typedef struct {
  int count[3][3][5][72][28];
  float mean[3][3][5][72][28];
  float stddev[3][3][5][72][28];
  int hist[30][3][3][5][72][28];
} L3PR_G1_RAINRATENEARSURFACE;

#endif

#ifndef _L3PR_G1_PRECIPRATENEARSURFACE_
#define _L3PR_G1_PRECIPRATENEARSURFACE_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_PRECIPRATENEARSURFACE;

#endif

#ifndef _L3PR_G1_PRECIPRATEESURFACE2_
#define _L3PR_G1_PRECIPRATEESURFACE2_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_PRECIPRATEESURFACE2;

#endif

#ifndef _L3PR_G1_PRECIPRATEESURFACE_
#define _L3PR_G1_PRECIPRATEESURFACE_

typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_PRECIPRATEESURFACE;

#endif

#ifndef _L3PR_G1_MIXEDPHRATE_
#define _L3PR_G1_MIXEDPHRATE_

typedef struct {
    int count[3][3][5][5][72][28];
    float mean[3][3][5][5][72][28];
    float stdev[3][3][5][5][72][28];
    int hist[30][3][3][5][5][72][28];
} L3PR_G1_MIXEDPHRATE;

#endif
typedef struct {
    int count[3][3][5][72][28];
    float mean[3][3][5][72][28];
    float stdev[3][3][5][72][28];
    int hist[30][3][3][5][72][28];
} L3PR_G1_FLAGHEAVYICEPRECIP;

#endif

#ifndef _L3PR_G1_SNOWRATE_
#define _L3PR_G1_SNOWRATE_

typedef struct {
    int count[3][3][5][5][72][28];
    float mean[3][3][5][5][72][28];
    float stdev[3][3][5][5][72][28];
    int hist[30][3][3][5][5][72][28];
} L3PR_G1_SNOWRATE;

#endif

#ifndef _L3PR_G1_RAINRATE_
#define _L3PR_G1_RAINRATE_

typedef struct {
    int count[3][3][5][5][72][28];
    float mean[3][3][5][5][72][28];
    float stdev[3][3][5][5][72][28];
    int hist[30][3][3][5][5][72][28];
} L3PR_G1_RAINRATE;

#endif

#ifndef _L3PR_G1_PRECIPRATE_
#define _L3PR_G1_PRECIPRATE_

typedef struct {
    int count[3][3][5][5][72][28];
    float mean[3][3][5][5][72][28];
    float stdev[3][3][5][5][72][28];
    int hist[30][3][3][5][5][72][28];
} L3PR_G1_PRECIPRATE;

#endif
typedef struct {
    int count[3][3][5][5][72][28];
    float mean[3][3][5][5][72][28];
    float stdev[3][3][5][5][72][28];
    int hist[30][3][3][5][5][72][28];
} L3PR_G1_PRECIPRATE;

#endif

#ifndef _L3PR_G1_
#define _L3PR_G1_

typedef struct {
    L3PR_G1_PRECIPRATE precipRate;
    L3PR_G1_RAINRATE rainRate;
    L3PR_G1_SNOWRATE snowRate;
    L3PR_G1_FLAGHEAVYICEPRECIP flagHeavyIcePrecip;
    L3PR_G1_MIXEDPHRATE mixedPhRate;
    L3PR_G1_PRECIPRATEESURFACE precipRateESurface;
    L3PR_G1_PRECIPRATEESURFACE2 precipRateESurface2;
    L3PR_G1_PRECIPRATENEARSURFACE precipRateNearSurface;
    L3PR_G1_RAINRATENEARSURFACE rainRateNearSurface;
    L3PR_G1_SNOWRATENEARSURFACE snowRateNearSurface;
    L3PR_G1_MIXEDPHRATENEARSURFACE mixedPhRateNearSurface;
    L3PR_G1_PRECIPWATERINTEGRATED precipWaterIntegrated;
    L3PR_G1_PRECIPICEINTEGRATED precipIceIntegrated;
    L3PR_G1_PRECIPRATEAVE24 precipRateAve24;
    L3PR_G1_ZFACTORCORRECTED zFactorCorrected;
    L3PR_G1_ZFACTORCORRECTEDESURFACE zFactorCorrectedESurface;
    L3PR_G1_ZFACTORCORRECTEDNEARSURFACE zFactorCorrectedNearSurface;
    L3PR_G1_ZFACTORMEASUREDNEARSURFACE zFactorMeasuredNearSurface;
    L3PR_G1_ZFACTORCORRECTEDDPR zFactorCorrectedDPR;
    L3PR_G1_ZFACTORCORRECTEDESURFACEDPR zFactorCorrectedESurfaceDPR;
    L3PR_G1_ZFACTORCORRECTEDNEARSURFACEDPR zFactorCorrectedNearSurfaceDPR;
    L3PR_G1_ZFACTORMEASURED zFactorMeasured;
    L3PR_G1_DM dm;
    L3PR_G1_DBNW dBNw;
    L3PR_G1_EPSILONDPR epsilonDPR;
    L3PR_G1_EPSILON epsilon;
    L3PR_G1_ZETA zeta;
    L3PR_G1_PIAHB piaHB;
    L3PR_G1_PIAHYBRID piaHybrid;
    L3PR_G1_PIAHYBRIIDPR piaHybridDPR;
};
L3PR_G1_PIASRT piaSRT;
L3PR_G1_PIASRTDPR piaSRTdpr;
L3PR_G1_PIAFINAL piaFinal;
L3PR_G1_PIAFINALLDPR piaFinalDPR;
L3PR_G1_PIAFINALLSUBSET piaFinalSubset;
L3PR_G1_PIAFINALLDPRSUBSET piaFinalDPRsubset;
L3PR_G1_HEIGHTBB heightBB;
L3PR_G1_HEIGHTBBNADIR heightBBnadir;
L3PR_G1_BBWIDTHNADIR BBwidthNadir;
L3PR_G1_HEIGHTSTORMTOP heightStormTop;
L3PR_G1_BBWIDTH BBwidth;
L3PR_G1_OBSERVATIONCOUNTS observationCounts;
L3PR_G1_PRECIPRATELOCALTIME precipRateLocalTime;
L3PR_G1_DFRMNEARSURFACE DFRmNearSurface;
L3PR_G1_DFRMNEARSURFACE DFRNearSurface;
float precipRateNearSurfaceUnconditional[5][72][28];
float precipProbabilityNearSurface[5][72][28];
} L3PR_G1;

#endif

#ifndef _L3PR_GRIDS_
#define _L3PR_GRIDS_

typedef struct {
    L3PR_G1 G1;
    L3PR_G2 G2;
} L3PR_GRIDS;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L3PR_G2_DFRMNEARSURFACE/
    INTEGER*4 count(536,1440,3)
    REAL*4 mean(536,1440,3)
    REAL*4 stdev(536,1440,3)
END STRUCTURE

STRUCTURE /L3PR_G2_DFRMNEARSURFACE/
    INTEGER*4 count(536,1440,3)
REAL*4 mean(536,1440,3)
REAL*4 stdev(536,1440,3)
END STRUCTURE

STRUCTURE /L3PR_G2_OBSERVATIONCOUNTS/
INTEGER*4 total(536,1440,4)
INTEGER*4 pia(536,1440,4,7)
INTEGER*4 shallowRain(536,1440,4)
END STRUCTURE

STRUCTURE /L3PR_G2_BBWIDTH/
INTEGER*4 count(536,1440,5,3)
REAL*4 mean(536,1440,5,3)
REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_HEIGHTSTORMTOP/
INTEGER*4 count(536,1440,5,3)
REAL*4 mean(536,1440,5,3)
REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_HEIGHTBB/
INTEGER*4 count(536,1440,5,3)
REAL*4 mean(536,1440,5,3)
REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PIAFINALDPR/
INTEGER*4 count(536,1440,4,7,3)
REAL*4 mean(536,1440,4,7,3)
REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PIAFINAL/
INTEGER*4 count(536,1440,4,7,3)
REAL*4 mean(536,1440,4,7,3)
REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PIASRTDPR/
INTEGER*4 count(536,1440,4,7,3)
REAL*4 mean(536,1440,4,7,3)
REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PIASRT/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PIAHYBRIDDPR/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PIAHYBRID/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PIAHB/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3PR_G2_ZETA/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3PR_G2_EPSILON/
    INTEGER*4 count(536,1440,4,3)
    REAL*4 mean(536,1440,4,3)
    REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3PR_G2_EPSILON_DPR/
    INTEGER*4 count(536,1440,4,5,3)
    REAL*4 mean(536,1440,4,5,3)
    REAL*4 stdev(536,1440,4,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_DBNW/
    INTEGER*4 count(536,1440,5,3)
    REAL*4 mean(536,1440,5,3)
    REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_DM/
    INTEGER*4 count(536,1440,5,3)
    REAL*4 mean(536,1440,5,3)
    REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_ZFACTORMEASURED/
    INTEGER*4 count(536,1440,4,5,3)
    REAL*4 mean(536,1440,4,5,3)
    REAL*4 stdev(536,1440,4,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_ZFACTORCORRECTEDNEARSURFACEDPR/
    INTEGER*4 count(536,1440,4,3)
    REAL*4 mean(536,1440,4,3)
    REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3PR_G2_ZFACTORCORRECTEDDESURFACEDPR/
    INTEGER*4 count(536,1440,4,3)
    REAL*4 mean(536,1440,4,3)
    REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3PR_G2_ZFACTORCORRECTEDDPR/
    INTEGER*4 count(536,1440,4,5,3)
    REAL*4 mean(536,1440,4,5,3)
    REAL*4 stdev(536,1440,4,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_ZFACTORMEASUREDNEARSURFACE/
    INTEGER*4 count(536,1440,4,3)
    REAL*4 mean(536,1440,4,3)
    REAL*4 stdev(536,1440,4,3)
END STRUCTURE
STRUCTURE /L3PR_G2_ZFACTORCORRECTEDNEARSURFACE/
    INTEGER*4 count(536,1440,4,3)
    REAL*4 mean(536,1440,4,3)
    REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3PR_G2_ZFACTORCORRECTEDDESURFACE/
    INTEGER*4 count(536,1440,4,3)
    REAL*4 mean(536,1440,4,3)
    REAL*4 stdev(536,1440,4,3)
END STRUCTURE

STRUCTURE /L3PR_G2_ZFACTORCORRECTED/
    INTEGER*4 count(536,1440,4,5,3)
    REAL*4 mean(536,1440,4,5,3)
    REAL*4 stdev(536,1440,4,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PRECIPRATEAVE24/
    INTEGER*4 count(536,1440,5,3)
    REAL*4 mean(536,1440,5,3)
    REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PRECIPICEINTEGRATED/
    INTEGER*4 count(536,1440,5,3)
    REAL*4 mean(536,1440,5,3)
    REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PRECIPWATERINTEGRATED/
    INTEGER*4 count(536,1440,5,3)
    REAL*4 mean(536,1440,5,3)
    REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_MIXEDPHRATENEARSURFACE/
    INTEGER*4 count(536,1440,5,3)
    REAL*4 mean(536,1440,5,3)
    REAL*4 stdev(536,1440,5,3)
END STRUCTURE
STRUCTURE /L3PR_G2_SNOWRATENEARSURFACE/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_RAINRATENEARSURFACE/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PRECIPRATENEARSURFACE/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PRECIPRATEESURFACE2/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PRECIPRATEESURFACE/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_MIXEDPHRATE/
  INTEGER*4 count(536,1440,5,5,3)
  REAL*4 mean(536,1440,5,5,3)
  REAL*4 stdev(536,1440,5,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_FLAGHEAVYICEPRECIP/
  INTEGER*4 count(536,1440,5,3)
  REAL*4 mean(536,1440,5,3)
  REAL*4 stdev(536,1440,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_SNOWRATE/
INTEGER*4 count(536,1440,5,5,3)
REAL*4 mean(536,1440,5,5,3)
REAL*4 stdev(536,1440,5,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_RAINRATE/
  INTEGER*4 count(536,1440,5,5,3)
  REAL*4 mean(536,1440,5,5,3)
  REAL*4 stdev(536,1440,5,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2_PRECIPRATE/
  INTEGER*4 count(536,1440,5,5,3)
  REAL*4 mean(536,1440,5,5,3)
  REAL*4 stdev(536,1440,5,5,3)
END STRUCTURE

STRUCTURE /L3PR_G2/
  RECORD /L3PR_G2_PRECIPRATE/ precipRate
  RECORD /L3PR_G2_RAINRATE/ rainRate
  RECORD /L3PR_G2_SNOWRATE/ snowRate
  RECORD /L3PR_G2_FLAGHEAVYICEPRECIP/ flagHeavyIcePrecip
  RECORD /L3PR_G2_MIXEDPHRATE/ mixedPhRate
  RECORD /L3PR_G2_PRECIPRATEESURFACE/ precipRateESurface
  RECORD /L3PR_G2_PRECIPRATEESURFACE2/ precipRateESurface2
  RECORD /L3PR_G2_PRECIPATENEARSSURFACE/ precipRateNearSurface
  RECORD /L3PR_G2_RAINATENEARSSURFACE/ rainRateNearSurface
  RECORD /L3PR_G2_SNOWATENEARSSURFACE/ snowRateNearSurface
  RECORD /L3PR_G2_MIXEDPHATENEARSSURFACE/ mixedPhRateNearSurface
  RECORD /L3PR_G2_PRECIPWATERINTEGRATED/ precipWaterIntegrated
  RECORD /L3PR_G2_PRECIPICEINTEGRATED/ precipIceIntegrated
  RECORD /L3PR_G2_PRECIPRATEAVE24/ precipRateAve24
  RECORD /L3PR_G2_ZFACTORCORRECTED/ zFactorCorrected
  RECORD /L3PR_G2_ZFACTORCORRECTEDESURFACE/ zFactorCorrectedESurface
  RECORD /L3PR_G2_ZFACTORCORRECTEDNEARSSURFACE/ zFactorCorrectedNearSurface
  RECORD /L3PR_G2_ZFACTORMEASUREDNEARSSURFACE/ zFactorMeasuredNearSurface
  RECORD /L3PR_G2_ZFACTORCORRECTEDDPR/ zFactorCorrectedDPR
  RECORD /L3PR_G2_ZFACTORCORRECTEDESURFACEDPR/ zFactorCorrectedESurfaceDPR
  RECORD /L3PR_G2_ZFACTORCORRECTEDNEARSSURFACEDPR/ zFactorCorrectedNearSurfaceDPR
  RECORD /L3PR_G2_ZFACTORMEASURED/ zFactorMeasured
  RECORD /L3PR_G2_DM/ dm
  RECORD /L3PR_G2_DBNW/ dBNw
  RECORD /L3PR_G2_EPSILONDPR/ epsilonDPR
RECORD /L3PR_G2_EPSILON/ epsilon
RECORD /L3PR_G2_ZETA/ zeta
RECORD /L3PR_G2_PIAHB/ piaHB
RECORD /L3PR_G2_PIAHYBRID/ piaHybrid
RECORD /L3PR_G2_PIAHYBRIDDPR/ piaHybridDPR
RECORD /L3PR_G2_PIASRT/ piaSRT
RECORD /L3PR_G2_PIASRTDPR/ piaSRTdpr
RECORD /L3PR_G2_PIAFINAL/ piaFinal
RECORD /L3PR_G2_PIAFINALDPR/ piaFinalDPR
RECORD /L3PR_G2_HEIGHTBB/ heightBB
RECORD /L3PR_G2_HEIGHTSTORMTOP/ heightStormTop
RECORD /L3PR_G2_BBWIDTH/ BBwidth
RECORD /L3PR_G2_OBSERVATIONCOUNTS/ observationCounts
RECORD /L3PR_G2_DFRMNEARSURFACE/ DFRmNearSurface
RECORD /L3PR_G2_DFRNEARSURFACE/ DFRNearSurface
REAL*4 precipRateNearSurfaceUnconditional(536,1440,5)
REAL*4 precipProbabilityNearSurface(536,1440,5)
END STRUCTURE

STRUCTURE /L3PR_G1_DFRNEARSURFACE/
  INTEGER*4 count(28,72,3,3)
  REAL*4 mean(28,72,3,3)
  REAL*4 stdev(28,72,3,3)
  INTEGER*4 hist(28,72,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_DFRMNEARSURFACE/
  INTEGER*4 count(28,72,3,3)
  REAL*4 mean(28,72,3,3)
  REAL*4 stdev(28,72,3,3)
  INTEGER*4 hist(28,72,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PRECIPRATELOCALTIME/
  INTEGER*4 count(28,72,5,24,3)
  REAL*4 mean(28,72,5,24,3)
  REAL*4 stdev(28,72,5,24,3)
END STRUCTURE

STRUCTURE /L3PR_G1_OBSERVATIONCOUNTS/
  INTEGER*4 total(28,72,4,3)
  INTEGER*4 localTime(28,72,4,24,3)
  INTEGER*4 pia(28,72,4,7,3)
INTEGER*4 shallowRain(28,72,4,3)
END STRUCTURE

STRUCTURE /L3PR_G1_BBWIDTH/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_HEIGHTSTORMTOP/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_BBWIDTHNADIR/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_HEIGHTBBNADIR/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_HEIGHTBB/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PIAFINALDPRSUBSET/
  INTEGER*4 count(28,72,4,7,3,3)
  REAL*4 mean(28,72,4,7,3,3)
  REAL*4 stdev(28,72,4,7,3,3)
  INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PIAFINAISUBSET/
   INTEGER*4 count(28,72,4,7,3,3)
   REAL*4 mean(28,72,4,7,3,3)
   REAL*4 stdev(28,72,4,7,3,3)
   INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PIAFINALDPR/
   INTEGER*4 count(28,72,4,7,3,3)
   REAL*4 mean(28,72,4,7,3,3)
   REAL*4 stdev(28,72,4,7,3,3)
   INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PIAFINAL/
   INTEGER*4 count(28,72,4,7,3,3)
   REAL*4 mean(28,72,4,7,3,3)
   REAL*4 stdev(28,72,4,7,3,3)
   INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PIASRTDPR/
   INTEGER*4 count(28,72,4,7,3,3)
   REAL*4 mean(28,72,4,7,3,3)
   REAL*4 stdev(28,72,4,7,3,3)
   INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PIASRT/
   INTEGER*4 count(28,72,4,7,3,3)
   REAL*4 mean(28,72,4,7,3,3)
   REAL*4 stdev(28,72,4,7,3,3)
   INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PIAHYBRIDDPR/
   INTEGER*4 count(28,72,4,7,3,3)
   REAL*4 mean(28,72,4,7,3,3)
   REAL*4 stdev(28,72,4,7,3,3)
   INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE
STRUCTURE /L3PR_G1_PIAHYBRID/
    INTEGER*4 count(28,72,4,7,3,3)
    REAL*4 mean(28,72,4,7,3,3)
    REAL*4 stdev(28,72,4,7,3,3)
    INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PIAHB/
    INTEGER*4 count(28,72,4,7,3,3)
    REAL*4 mean(28,72,4,7,3,3)
    REAL*4 stdev(28,72,4,7,3,3)
    INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_ZETA/
    INTEGER*4 count(28,72,4,7,3,3)
    REAL*4 mean(28,72,4,7,3,3)
    REAL*4 stdev(28,72,4,7,3,3)
    INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_EPSILON/
    INTEGER*4 count(28,72,4,3,3)
    REAL*4 mean(28,72,4,3,3)
    REAL*4 stdev(28,72,4,3,3)
    INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_EPSILONDPR/
    INTEGER*4 count(28,72,4,5,3,3)
    REAL*4 mean(28,72,4,5,3,3)
    REAL*4 stdev(28,72,4,5,3,3)
    INTEGER*4 hist(28,72,4,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_DBNW/
    INTEGER*4 count(28,72,5,3,3)
    REAL*4 mean(28,72,5,3,3)
    REAL*4 stdev(28,72,5,3,3)
    INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE
STRUCTURE /L3PR_G1_DM/
    INTEGER*4  count(28,72,5,3,3)
    REAL*4    mean(28,72,5,3,3)
    REAL*4    stdev(28,72,5,3,3)
    INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_ZFACTORMEASURED/
    INTEGER*4  count(28,72,4,5,3,3)
    REAL*4    mean(28,72,4,5,3,3)
    REAL*4    stdev(28,72,4,5,3,3)
    INTEGER*4 hist(28,72,4,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_ZFACTORCORRECTEDNEARSURFACE_DPR/
    INTEGER*4  count(28,72,4,3,3)
    REAL*4    mean(28,72,4,3,3)
    REAL*4    stdev(28,72,4,3,3)
    INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_ZFACTORCORRECTEDDESURFACE_DPR/
    INTEGER*4  count(28,72,4,3,3)
    REAL*4    mean(28,72,4,3,3)
    REAL*4    stdev(28,72,4,3,3)
    INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_ZFACTORCORRECTEDDPR/
    INTEGER*4  count(28,72,4,5,3,3)
    REAL*4    mean(28,72,4,5,3,3)
    REAL*4    stdev(28,72,4,5,3,3)
    INTEGER*4 hist(28,72,4,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_ZFACTORMEASUREDNEARSURFACE/
    INTEGER*4  count(28,72,4,3,3)
    REAL*4    mean(28,72,4,3,3)
    REAL*4    stdev(28,72,4,3,3)
    INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_ZFACTORCORRECTEDNEARSURFACE/
INTEGER*4 count(28,72,4,3,3)
REAL*4 mean(28,72,4,3,3)
REAL*4 stdev(28,72,4,3,3)
INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_ZFACTORCORRECTEDESURFACE/
   INTEGER*4 count(28,72,4,3,3)
   REAL*4 mean(28,72,4,3,3)
   REAL*4 stdev(28,72,4,3,3)
   INTEGER*4 hist(28,72,4,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_ZFACTORCORRECTED/
   INTEGER*4 count(28,72,4,5,3,3)
   REAL*4 mean(28,72,4,5,3,3)
   REAL*4 stdev(28,72,4,5,3,3)
   INTEGER*4 hist(28,72,4,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PRECIPRATEAVE24/
   INTEGER*4 count(28,72,5,3,3)
   REAL*4 mean(28,72,5,3,3)
   REAL*4 stdev(28,72,5,3,3)
   INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PRECIPICEINTEGRATED/
   INTEGER*4 count(28,72,5,3,3)
   REAL*4 mean(28,72,5,3,3)
   REAL*4 stdev(28,72,5,3,3)
   INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_MIXEDPHRATENEARSURFACE/
   INTEGER*4 count(28,72,5,3,3)
REAL*4 mean(28,72,5,3,3)
REAL*4 stdev(28,72,5,3,3)
INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_SNOWRATENEARSURFACE/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_RAINRATENEARSURFACE/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PRECIPRATENEARSURFACE/
  INTEGER*4 count(28,72,5,3,3)
  REAL*4 mean(28,72,5,3,3)
  REAL*4 stdev(28,72,5,3,3)
  INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_MIXEDPHRATE/
  INTEGER*4 count(28,72,5,5,3,3)
  REAL*4 mean(28,72,5,3,3)
END STRUCTURE
REAL*4 stdev(28,72,5,3,3)
INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_FLAGHEAVYICEPRECIP/
   INTEGER*4 count(28,72,5,3,3)
   REAL*4 mean(28,72,5,3,3)
   REAL*4 stdev(28,72,5,3,3)
   INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_SNOWRATE/
   INTEGER*4 count(28,72,5,3,3)
   REAL*4 mean(28,72,5,3,3)
   REAL*4 stdev(28,72,5,3,3)
   INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_RAINRATE/
   INTEGER*4 count(28,72,5,3,3)
   REAL*4 mean(28,72,5,3,3)
   REAL*4 stdev(28,72,5,3,3)
   INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1_PRECIPRATE/
   INTEGER*4 count(28,72,5,3,3)
   REAL*4 mean(28,72,5,3,3)
   REAL*4 stdev(28,72,5,3,3)
   INTEGER*4 hist(28,72,5,3,3,30)
END STRUCTURE

STRUCTURE /L3PR_G1/
   RECORD /L3PR_G1_PRECIPRATE/ precipRate
   RECORD /L3PR_G1_RAINRATE/ rainRate
   RECORD /L3PR_G1_SNOWRATE/ snowRate
   RECORD /L3PR_G1_FLAGHEAVYICEPRECIP/ flagHeavyIcePrecip
   RECORD /L3PR_G1_MIXEDPHRATE/ mixedPhRate
   RECORD /L3PR_G1_PRECIPRATEESURFACE/ precipRateESurface
   RECORD /L3PR_G1_PRECIPRATEESURFACE2/ precipRateESurface2
   RECORD /L3PR_G1_PRECIPRATENEARSURFACE/ precipRateNearSurface
   RECORD /L3PR_G1_RAINRATENEARSURFACE/ rainRateNearSurface
   RECORD /L3PR_G1_SNOWRATENEARSURFACE/ snowRateNearSurface
RECORD /L3PR_G1_MIXEDPHRATENEARSURFACE/ mixedPhRateNearSurface
RECORD /L3PR_G1_PRECIPWATERINTEGRATED/ precipWaterIntegrated
RECORD /L3PR_G1_PRECIPICEINTEGRATED/ precipIceIntegrated
RECORD /L3PR_G1_PRECIPRATEAVE24/ precipRateAve24
RECORD /L3PR_G1_ZFACTORCORRECTED/ zFactorCorrected
RECORD /L3PR_G1_ZFACTORCORRECTEDESURFACE/ zFactorCorrectedESurface
RECORD /L3PR_G1_ZFACTORCORRECTEDNEARSURFACE/ zFactorCorrectedNearSurface
RECORD /L3PR_G1_ZFACTORMEASUREDNEARSURFACE/ zFactorMeasuredNearSurface
RECORD /L3PR_G1_ZFACTORCORRECTEDDPR/ zFactorCorrectedDPR
RECORD /L3PR_G1_ZFACTORCORRECTEDESURFACEDPR/ zFactorCorrectedESurfaceDPR
RECORD /L3PR_G1_ZFACTORCORRECTEDNEARSURFACEDPR/ zFactorCorrectedNearSurfaceDPR
RECORD /L3PR_G1_ZFACTORMEASURED/ zFactorMeasured
RECORD /L3PR_G1_DM/ dm
RECORD /L3PR_G1_DBNW/ dBNw
RECORD /L3PR_G1_EPSILON/ epsilon
RECORD /L3PR_G1_EPSILONON/ epsilonDPR
RECORD /L3PR_G1_ZETA/ zeta
RECORD /L3PR_G1_PIAHB/ piaHB
RECORD /L3PR_G1_PIAHYBRID/ piaHybrid
RECORD /L3PR_G1_PIAHYBRIDDPR/ piaHybridDPR
RECORD /L3PR_G1_PIASRT/ piaSRT
RECORD /L3PR_G1_PIASRTDPR/ piaSRTdpr
RECORD /L3PR_G1_PIAFINAL/ piaFinal
RECORD /L3PR_G1_PIAFINALDPR/ piaFinalDPR
RECORD /L3PR_G1_PIAFINALSUBSET/ piaFinalSubset
RECORD /L3PR_G1_PIAFINALDPRSUBSET/ piaFinalDPRsubset
RECORD /L3PR_G1_HEIGHTBB/ heightBB
RECORD /L3PR_G1_HEIGHTBBNADIR/ heightBBnadir
RECORD /L3PR_G1_BBWIDTH/ BBwidth
RECORD /L3PR_G1_BBWIDTHNADIR/ BBwidthNadir
RECORD /L3PR_G1_HEIGHTSTORMTOP/ heightStormTop
RECORD /L3PR_G1_OBSERVATIONCOUNTS/ observationCounts
RECORD /L3PR_G1_PRECIPRATELOCALTIME/ precipRateLocalTime
RECORD /L3PR_G1_DFRMNEARSURFACE/ DFRmNearSurface
RECORD /L3PR_G1_DFRNEARSURFACE/ DFRNearSurface
REAL*4 precipRateNearSurfaceUnconditional(28,72,5)
REAL*4 precipProbabilityNearSurface(28,72,5)
END STRUCTURE

STRUCTURE /L3PR_GRIDS/
  RECORD /L3PR_G1/ G1
  RECORD /L3PR_G2/ G2
END STRUCTURE
5.55 3PRD - PR Daily Product

3PRD, "PR Daily Product", computes daily statistics of the PR measurements at a high horizontal resolution (0.25° x 0.25° latitude/longitude).

Ranges and descriptions not included in this version.

Dimension definitions:

nlat  536  Number of high resolution 0.25° grid intervals of latitude from 67°S to 67°N.

nlon  1440 Number of high resolution 0.25° grid intervals of longitude from 180°W to 180°E.

nalt  5  Number of heights above the earth ellipsoid: 2km, 4km, 6km, 10km, and 15km.

nvar  3  Number of phase bins. Bins are counts of phase less than 100, counts of phase greater than or equal to 100 and less than 200, counts of phase greater than or equal to 200.

chn   2  Number of channels.

AD    2  Ascending or descending half of the orbit.

Figure 787 through Figure 790 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileVersion** (Metadata):
FileVersion contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputFileNames** (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

**InputAlgorithmVersions** (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

**InputGenerationDateTimes** (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.
5.55 3PRD - PR Daily Product

Figure 787: Data Format Structure for 3PRD, PR Daily Product
continued from last figure

GRID

- convPrecipRateESurfMean 4 bytes Array: nlat x nlon x chn x AD
- convPrecipPixNearSurf 2 bytes Array: nlat x nlon x chn x AD
- stratPrecipRateMean 4 bytes Array: nlat x nlon x nalt x chn x AD
- stratPrecipRateNearSurfMean 4 bytes Array: nlat x nlon x chn x AD
- stratPrecipRateESurfMean 4 bytes Array: nlat x nlon x chn x AD
- stratPrecipPixNearSurf 2 bytes Array: nlat x nlon x chn x AD
- bbHtMean 4 bytes Array: nlat x nlon x chn x AD
- stormHtMean 4 bytes Array: nlat x nlon x chn x AD
- phase 2 bytes Array: nlat x nlon x nalt x nvar x chn x AD
- phaseNearSurf 2 bytes Array: nlat x nlon x nvar x chn x AD

GridTimeAsc Group

GridTimeDes Group

Figure 788: Data Format Structure for 3PRD, PR Daily Product

GridTimeAsc

- Year 2 bytes Array: nlat x nlon
- Month 1 byte Array: nlat x nlon
- DayOfMonth 1 byte Array: nlat x nlon
- Hour 1 byte Array: nlat x nlon
- Minute 1 byte Array: nlat x nlon
- Second 1 byte Array: nlat x nlon
- MilliSecond 2 bytes Array: nlat x nlon
- DayOfYear 2 bytes Array: nlat x nlon

Figure 789: Data Format Structure for 3PRD, GridTimeAsc
GRID (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
Mean Precipitation rate, includes both liquid and solid phases at various height levels. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

rainRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
Mean rainfall rate, excludes solid precipitation at various height levels. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

mixedRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
Mean rainfall rate of the mixed phase precipitation at various height levels. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

snowRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
Mean rainfall rate of solid precipitation at various height levels. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateNearSurfMean (4-byte float, array size: nlat x nlon x chn x AD):

Figure 790: Data Format Structure for 3PRD, GridTimeDes
Mean precipitation rate in a grid box using only the Near Surface location along the slant path for each radar ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**rainRateNearSurfMean** (4-byte float, array size: nlat x nlon x chn x AD):
Mean rainfall rate of liquid precipitation in a grid box using only the Near Surface location along the slant path for each radar ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**mixedRateNearSurfMean** (4-byte float, array size: nlat x nlon x chn x AD):
Mean rainfall rate of mixed phase precipitation in a grid box using only the Near Surface location along the slant path for each radar ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**snowRateNearSurfMean** (4-byte float, array size: nlat x nlon x chn x AD):
Mean rainfall rate of solid precipitation in a grid box using only the Near Surface location along the slant path for each radar ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurfMean** (4-byte float, array size: nlat x nlon x chn x AD):
Mean precipitation rate in a grid box using only the Estimated Surface location along the slant path for each radar ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurf2Mean** (4-byte float, array size: nlat x nlon x chn x AD):
Mean precipitation rate in a grid box using only the Estimated Surface 2 location along the slant path for each radar ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**totalPix** (2-byte integer, array size: nlat x nlon x chn x AD):
The total number of measurements in each grid box. First index is Ascending node, second index is Descending. Special values are defined as:
-9999 Missing value

**precipPix** (2-byte integer, array size: nlat x nlon x nalt x chn x AD):
The number of measurements in each grid box that included detectable precipitation at various height levels. First index is Ascending node, second index is Descending. Special values are defined as:
-9999 Missing value

**precipPixNearSurf** (2-byte integer, array size: nlat x nlon x chn x AD):
The number of measurements in a grid box that included detectable precipitation at the Near Surface level. First index is Ascending node, second index is Descending. Special
values are defined as:
  -9999  Missing value

precipPixESurf (2-byte integer, array size: nlat x nlon x chn x AD):
The number of measurements in a grid box that included detectable precipitation at
the Estimated Surface level. First index is Ascending node, second index is Descending.
Special values are defined as:
  -9999  Missing value

convPrecipRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
The mean precipitation rate of convective type at various height levels. First index is
Ascending node, second index is Descending. Values are in mm/hr. Special values are
defined as:
  -9999.9  Missing value

convPrecipRateNearSurfMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean precipitation rate of convective type at the Near Surface level along the radar
ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are
defined as:
  -9999.9  Missing value

convPrecipRateESurfMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean precipitation rate of convective type at the Estimated Surface level along the
radar ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are
defined as:
  -9999.9  Missing value

convPrecipPixNearSurf (2-byte integer, array size: nlat x nlon x chn x AD):
The number of convective precipitation measurements in a grid box at the Near Surface
level. First index is Ascending node, second index is Descending. Special values are
defined as:
  -9999  Missing value

stratPrecipRateMean (4-byte float, array size: nlat x nlon x nalt x chn x AD):
The mean precipitation rate of stratiform type at various height levels. First index is
Ascending node, second index is Descending. Values are in mm/hr. Special values are
defined as:
  -9999.9  Missing value

stratPrecipRateNearSurfMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean precipitation rate of stratiform type at the Near Surface level along the radar
ray. First index is Ascending node, second index is Descending. Values are in mm/hr. Special values are
defined as:
  -9999.9  Missing value

stratPrecipRateESurfMean (4-byte float, array size: nlat x nlon x chn x AD):
The mean precipitation rate of stratiform type at the Estimated Surface level along the
radar ray. First index is Ascending node, second index is Descending. Values are in mm/hr.
mm/hr. Special values are defined as:
-9999.9 Missing value

**stratPrecipPixNearSurf** (2-byte integer, array size: nlat x nlon x chn x AD):
The number of stratiform precipitation measurements in a grid box at the Near Surface level. First index is Ascending node, second index is Descending. Special values are defined as:
-9999 Missing value

**bbHtMean** (4-byte float, array size: nlat x nlon x chn x AD):
The mean bright band height in a grid box. First index is Ascending node, second index is Descending. Values are in m. Special values are defined as:
-9999.9 Missing value

**stormHtMean** (4-byte float, array size: nlat x nlon x chn x AD):
The mean storm height in a grid box. First index is Ascending node, second index is Descending. Values are in m. Special values are defined as:
-9999.9 Missing value

**phase** (2-byte integer, array size: nlat x nlon x nalt x nvar x chn x AD):
The precipitation phase type in a grid box at various heights. First index is Ascending node, second index is Descending. Special values are defined as:
-9999 Missing value

**phaseNearSurf** (2-byte integer, array size: nlat x nlon x nvar x chn x AD):
The precipitation phase type in a grid box. First index is Ascending node, second index is Descending. Special values are defined as:
-9999 Missing value

**GridTimeAsc** (Group)
A UTC time associated with the grid box.

**Year** (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

GridTimeDes (Group)
A UTC time associated with the grid box.

Year (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value
DayOfYear (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

C Structure Header file:

```c
#ifndef _TK_3PRD_H_
#define _TK_3PRD_H_

#ifndef _L3PRD_GRIDTIMEDES_
#define _L3PRD_GRIDTIMEDES_

typedef struct {
    short Year[1440][536];
    signed char Month[1440][536];
    signed char DayOfMonth[1440][536];
    signed char Hour[1440][536];
    signed char Minute[1440][536];
    signed char Second[1440][536];
    short MilliSecond[1440][536];
    short DayOfYear[1440][536];
} L3PRD_GRIDTIMEDES;

#endif

#ifndef _L3PRD_GRIDTIMEASC_
#define _L3PRD_GRIDTIMEASC_

typedef struct {
    short Year[1440][536];
    signed char Month[1440][536];
    signed char DayOfMonth[1440][536];
    signed char Hour[1440][536];
    signed char Minute[1440][536];
    signed char Second[1440][536];
    short MilliSecond[1440][536];
    short DayOfYear[1440][536];
} L3PRD_GRIDTIMEASC;

#endif

#ifndef _L3PRD_GRID_
#define _L3PRD_GRID_
```

typedef struct {
    float precipRateMean[2][2][5][1440][536];
    float rainRateMean[2][2][5][1440][536];
    float mixedRateMean[2][2][5][1440][536];
    float snowRateMean[2][2][5][1440][536];
    float precipRateNearSurfMean[2][2][1440][536];
    float rainRateNearSurfMean[2][2][1440][536];
    float mixedRateNearSurfMean[2][2][1440][536];
    float snowRateNearSurfMean[2][2][1440][536];
    float precipRateESurfMean[2][2][1440][536];
    float precipRateESurf2Mean[2][2][1440][536];
    short totalPix[2][2][1440][536];
    short precipPix[2][2][5][1440][536];
    short precipPixNearSurf[2][2][1440][536];
    short precipPixESurf[2][2][1440][536];
    float convPrecipRateMean[2][2][5][1440][536];
    float convPrecipRateNearSurfMean[2][2][1440][536];
    float convPrecipRateESurfMean[2][2][1440][536];
    short convPrecipPixNearSurf[2][2][1440][536];
    float stratPrecipRateMean[2][2][5][1440][536];
    float stratPrecipRateNearSurfMean[2][2][1440][536];
    float stratPrecipRateESurfMean[2][2][1440][536];
    short stratPrecipPixNearSurf[2][2][1440][536];
    float bbHtMean[2][2][1440][536];
    float stormHtMean[2][2][1440][536];
    short phase[2][2][3][5][1440][536];
    short phaseNearSurf[2][2][3][1440][536];
    L3PRD_GRIDTIMEASC GridTimeAsc;
    L3PRD_GRIDTIMEDES GridTimeDes;
} L3PRD_GRID;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L3PRD_GRIDTIMEDES/
    INTEGER*2 Year(536,1440)
    BYTE Month(536,1440)
    BYTE DayOfMonth(536,1440)
    BYTE Hour(536,1440)
    BYTE Minute(536,1440)
BYTE  Second(536,1440)
INTEGER*2  MilliSecond(536,1440)
INTEGER*2  DayOfYear(536,1440)
END STRUCTURE

STRUCTURE  /L3PRD_GRIDTIMEASC/
  INTEGER*2  Year(536,1440)
  BYTE  Month(536,1440)
  BYTE  DayOfMonth(536,1440)
  BYTE  Hour(536,1440)
  BYTE  Minute(536,1440)
  BYTE  Second(536,1440)
  INTEGER*2  MilliSecond(536,1440)
  INTEGER*2  DayOfYear(536,1440)
END STRUCTURE

STRUCTURE  /L3PRD_GRID/
  REAL*4  precipRateMean(536,1440,5,2,2)
  REAL*4  rainRateMean(536,1440,5,2,2)
  REAL*4  mixedRateMean(536,1440,5,2,2)
  REAL*4  snowRateMean(536,1440,5,2,2)
  REAL*4  precipRateNearSurfMean(536,1440,5,2,2)
  REAL*4  rainRateNearSurfMean(536,1440,5,2,2)
  REAL*4  mixedRateNearSurfMean(536,1440,5,2,2)
  REAL*4  snowRateNearSurfMean(536,1440,5,2,2)
  REAL*4  precipRateESurfMean(536,1440,5,2,2)
  REAL*4  precipRateESurf2Mean(536,1440,5,2,2)
  REAL*4  convPrecipRateMean(536,1440,5,2,2)
  REAL*4  convPrecipRateNearSurfMean(536,1440,5,2,2)
  REAL*4  convPrecipRateESurfMean(536,1440,5,2,2)
  REAL*4  stratPrecipRateMean(536,1440,5,2,2)
  REAL*4  stratPrecipRateNearSurfMean(536,1440,5,2,2)
  REAL*4  stratPrecipRateESurfMean(536,1440,5,2,2)
  REAL*4  bbHtMean(536,1440,5,2,2)
  REAL*4  stormHtMean(536,1440,5,2,2)
  INTEGER*2  phase(536,1440,5,3,2,2)
  INTEGER*2  phaseNearSurf(536,1440,5,3,2,2)
The Combined Level-2 product, 2BCMB, "Level-2 DPR and GMI Combined," is written as a two-swath structure. The first swath, NS, contains 49 rays that match Ku DPR. The second swath, MS, contains 25 rays that match Ka Matched DPR. Surface variables refer to the level of the 2ADPR "near surface", not the "estimated surface". The following sections describe the structure and contents of the format.

Dimension definitions:

- \texttt{nscan} \hspace{5mm} \text{var}  \hspace{5mm} \text{Number of scans in the granule.}
- \texttt{nrayNS} \hspace{5mm} 49 \hspace{5mm} \text{Number of rays (angle bins) in each NS scan.}
- \texttt{nrayMS} \hspace{5mm} 25 \hspace{5mm} \text{Number of rays (angle bins) in each MS scan.}
- \texttt{nBnEnv} \hspace{5mm} 10 \hspace{5mm} \text{Number of environmental bins.}
- \texttt{nBnPSDlo} \hspace{5mm} 9 \hspace{5mm} \text{Number of low resolution vertical range bins. The bin indices of the low resolution PSD profile parameters are found in PSDparam-LowNode.}
- \texttt{nBnPSDhi} \hspace{5mm} 88 \hspace{5mm} \text{Number of high resolution vertical range bins at 250m interval.}
- \texttt{nPSDlo} \hspace{5mm} 2 \hspace{5mm} \text{Number of low resolution precipitation drop-size distribution parameters. Parameters are log10(Nw), mu.}
- \texttt{nPSDhi} \hspace{5mm} 1 \hspace{5mm} \text{Number of high resolution precipitation drop-size distribution parameters.}
- \texttt{nBnTrBnd} \hspace{5mm} 2 \hspace{5mm} \text{Number of bins in phase transition boundary.}
- \texttt{nBnTr} \hspace{5mm} 10 \hspace{5mm} \text{Number of bins in phase transition.}
- \texttt{nPhsBnN} \hspace{5mm} 5 \hspace{5mm} \text{Number of phase bin nodes.}
- \texttt{nAB} \hspace{5mm} 2 \hspace{5mm} \text{Number of power law parameters. These parameters describe particle density. The parameters are alpha and beta.}
- \texttt{nemiss} \hspace{5mm} 13 \hspace{5mm} \text{Number of microwave surface emissivities for GMI channels, including separate emissivities for the double side-band channels.}
- \texttt{nKuKa} \hspace{5mm} 2 \hspace{5mm} \text{Number of Ku and Ka}
- \texttt{ncomp} \hspace{5mm} 5 \hspace{5mm} \text{Maximum number principal components (prinComp) stored for a given observed reflectivity profile.}

Figure 791 through Figure 807 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 791: Data Format Structure for 2BCMB, Level-2 DPR and GMI Combined
### NS_SwathHeader
- **ScanTime**: 19 bytes  
  - Group: nscan
- **Latitude**: 4 bytes  
  - Array: nrayNS x nscan
- **Longitude**: 4 bytes  
  - Array: nrayNS x nscan
- **scanStatus**: 24 bytes  
  - Group: nscan
- **navigation**: 84 bytes  
  - Group: nscan
- **Input**: 3038 bytes  
  - Group: nscan
- **aPriori**: 3136 bytes  
  - Group: nscan
- **surfaceAirPressure**: 4 bytes  
  - Array: nrayNS x nscan
- **surfaceAirTemperature**: 4 bytes  
  - Array: nrayNS x nscan
- **surfaceVaporDensity**: 4 bytes  
  - Array: nrayNS x nscan
- **skinTemperature**: 4 bytes  
  - Array: nrayNS x nscan
- **envParamNode**: 2 bytes  
  - Array: nBnEnv x nrayNS x nscan
- **airPressure**: 4 bytes  
  - Array: nBnEnv x nrayNS x nscan
- **airTemperature**: 4 bytes  
  - Array: nBnEnv x nrayNS x nscan
- **vaporDensity**: 4 bytes  
  - Array: nBnEnv x nrayNS x nscan
- **cloudLiqWaterCont**: 4 bytes  
  - Array: nBnPSDhi x nrayNS x nscan
- **cloudIceWaterCont**: 4 bytes  
  - Array: nBnPSDhi x nrayNS x nscan
- **phaseBinNodes**: 2 bytes  
  - Array: nPhsBnN x nrayNS x nscan
- **PSDparamLowNode**: 2 bytes  
  - Array: nBnPSDlo x nrayNS x nscan
- **precipTotPSDparamLow**: 4 bytes  
  - Array: nPSDlo x nBnPSDlo x nrayNS x nscan
- **precipTotPSDparamHigh**: 4 bytes  
  - Array: nBnPSDhi x nrayNS x nscan
- **precipTotWaterCont**: 4 bytes  
  - Array: nBnPSDhi x nrayNS x nscan
- **precipTotWaterContSigma**: 4 bytes  
  - Array: nBnPSDhi x nrayNS x nscan

continued on next figure

Figure 792: Data Format Structure for 2BCMB, NS, NS
continued from last figure

<table>
<thead>
<tr>
<th>Variable</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>precipTotRate</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>precipTotRateSigma</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>liqMassFracTrans</td>
<td>4 bytes</td>
<td>Array: nBnTr x nrayNS x nscan</td>
</tr>
<tr>
<td>liqRateFracTrans</td>
<td>4 bytes</td>
<td>Array: nBnTr x nrayNS x nscan</td>
</tr>
<tr>
<td>surfPrecipTotRate</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>surfPrecipTotRateSigma</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>surfLiqRateFrac</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>tenMeterWindSpeed</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>surfEmissivity</td>
<td>4 bytes</td>
<td>Array: nemiss x nrayNS x nscan</td>
</tr>
<tr>
<td>simulatedBrightTemp</td>
<td>4 bytes</td>
<td>Array: nemiss x nrayNS x nscan</td>
</tr>
<tr>
<td>nubfPIAfactor</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>multiScatMaxContrib</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>surfEmissSigma</td>
<td>4 bytes</td>
<td>Array: nemiss x nrayNS x nscan</td>
</tr>
<tr>
<td>tenMeterWindSigma</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>skinTempSigma</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>columnVaporSigma</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>columnCloudLiqSigma</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>errorOfDataFit</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>pia</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>CorrectedReflectFactor</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>FLG</td>
<td>588 bytes</td>
<td>Group: nscan</td>
</tr>
</tbody>
</table>

Figure 793: Data Format Structure for 2BCMB, NS
Figure 794: Data Format Structure for 2BCMB, MS, MS
continued from last figure

- precipTotRate: 4 bytes, Array: nBnPSDhi x nrayMS x nscan
- precipTotRateSigma: 4 bytes, Array: nBnPSDhi x nrayMS x nscan
- liqMassFracTrans: 4 bytes, Array: nBnTr x nrayMS x nscan
- liqRateFracTrans: 4 bytes, Array: nBnTr x nrayMS x nscan
- surfPrecipTotRate: 4 bytes, Array: nrayMS x nscan
- surfPrecipTotRateSigma: 4 bytes, Array: nrayMS x nscan
- surfLiqRateFrac: 4 bytes, Array: nrayMS x nscan
- tenMeterWindSpeed: 4 bytes, Array: nrayMS x nscan
- surfEmissivity: 4 bytes, Array: nemiss x nrayMS x nscan
- simulatedBrightTemp: 4 bytes, Array: nemiss x nrayMS x nscan
- nubfPIAfactor: 4 bytes, Array: nrayMS x nscan
- multiScatMaxContrib: 4 bytes, Array: nrayMS x nscan
- surfEmissSigma: 4 bytes, Array: nemiss x nrayMS x nscan
- tenMeterWindSigma: 4 bytes, Array: nrayMS x nscan
- skinTempSigma: 4 bytes, Array: nrayMS x nscan
- columnVaporSigma: 4 bytes, Array: nrayMS x nscan
- columnCloudLiqSigma: 4 bytes, Array: nrayMS x nscan
- errorOfDataFit: 4 bytes, Array: nrayMS x nscan
- pia: 4 bytes, Array: nKuKa x nrayMS x nscan
- correctedReflectFactor: 4 bytes, Array: nKuKa x nBnPSDhi x nrayMS x nscan
- FLG: 300 bytes, Group: nscan

Figure 795: Data Format Structure for 2BCMB, MS
5.56 2BCMB - Level-2 DPR and GMI Combined

ScanTime
- Year 2 bytes Array: nscan
- Month 1 byte Array: nscan
- DayOfMonth 1 byte Array: nscan
- Hour 1 byte Array: nscan
- Minute 1 byte Array: nscan
- Second 1 byte Array: nscan
- MilliSecond 2 bytes Array: nscan
- DayOfYear 2 bytes Array: nscan
- SecondOfDay 8 bytes Array: nscan

Figure 796: Data Format Structure for 2BCMB, NS, ScanTime

scanStatus
- dataQuality 1 byte Array: nscan
- dataWarning 1 byte Array: nscan
- missing 1 byte Array: nscan
- modeStatus 1 byte Array: nscan
- geoError 2 bytes Array: nscan
- geoWarning 2 bytes Array: nscan
- SCorientation 2 bytes Array: nscan
- pointingStatus 2 bytes Array: nscan
- acsModeMidScan 1 byte Array: nscan
- targetSelectionMidScan 1 byte Array: nscan
- operationalMode 1 byte Array: nscan
- limitErrorFlag 1 byte Array: nscan
- FractionalGranuleNumber 8 bytes Array: nscan

Figure 797: Data Format Structure for 2BCMB, NS, scanStatus
Figure 798: Data Format Structure for 2BCMB, NS, navigation
Figure 799: Data Format Structure for 2BCMB, NS, Input

Figure 800: Data Format Structure for 2BCMB, NS, aPriori

Figure 801: Data Format Structure for 2BCMB, NS, FLG
Figure 802: Data Format Structure for 2BCMB, MS, ScanTime

Figure 803: Data Format Structure for 2BCMB, MS, scanStatus
Figure 804: Data Format Structure for 2BCMB, MS, navigation
Figure 805: Data Format Structure for 2BCMB, MS, Input

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
5.56  2BCMB - Level-2 DPR and GMI Combined

**aPriori**
- **profClass** 4 bytes, Array: nrayMS x nscan
- ** prinComp** 4 bytes, Array: ncomp x nrayMS x nscan
- **surfPrecipBiasRatio** 4 bytes, Array: nrayMS x nscan
- **initNw** 4 bytes, Array: nBnPSDlo x nrayMS x nscan

*Figure 806: Data Format Structure for 2BCMB, MS, aPriori*

**FLG**
- **ioQuality** 4 bytes, Array: nrayMS x nscan
- ** multiScatCalc** 4 bytes, Array: nrayMS x nscan
- ** algoType** 4 bytes, Array: nrayMS x nscan

*Figure 807: Data Format Structure for 2BCMB, MS, FLG*

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**NS** (Swath)

**NS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in NS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: nrayNS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nrayNS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in NS)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

```
Bit Meaning if bit = 1
0  missing
5  geoError is not zero
6  modeStatus is not zero
```
**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

- Bit Meaning if bit = 1
  - 0 Beam matching is abnormal
  - 1 VPRF table is abnormal
  - 2 Surface table is abnormal
  - 3 geoWarning is not zero
  - 4 Operational mode is not observation mode
  - 5 GPS status is abnormal
  - 6 Spare (always 0)
  - 7 Check sum of L1A is abnormal

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

- Bit Meaning if bit = 1
  - 0 Scan is missing
  - 1 Science telemetry packet missing
  - 2 Science telemetry segment within packet missing
  - 3 Science telemetry other missing
  - 4 Housekeeping (HK) telemetry packet missing
  - 5 Spare (always 0)
  - 6 Spare (always 0)
  - 7 Spare (always 0)

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

- Bit Meaning if bit = 1
  - 0 Spare (always 0)
  - 1 SCorientation not 0 or 180
  - 2 pointingStatus not 0
  - 3 Non-routine limitErrorFlag
  - 4 Non-routine operationalMode (not 1 or 11)
  - 5 Spare (always 0)
  - 6 Spare (always 0)
  - 7 Spare (always 0)
**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
</tbody>
</table>
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**acsModeMidScan** (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENUL</td>
</tr>
</tbody>
</table>
2   SUNPOINT
3   GSPM (Gyro-less Sun Point)
4   MSM (Mission Science Mode)
5   SLEW
6   DELTAH
7   DELTAV
-99  UNKNOWN -- ACS mode unavailable

**targetSelectionMidScan** (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
</tbody>
</table>
17  Ku/Ka Independent Standby VPRF Table OUT
18  Ku/Ka Independent Standby Phase Out
19  Ku/Ka Independent Standby Dump Out
20  Ku/Ka Independent Standby (No Science Data)

`limitErrorFlag` (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. `limitErrorFlag` may be used in `modeStatus`. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

`FractionalGranuleNumber` (8-byte float, array size: nscan):

The floating point granule number. The granule begins at the Southern-most point of the spacecraft's trajectory. For example, `FractionalGranuleNumber = 10.5` means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

`scPos` (4-byte float, array size: XYZ x nscan):

The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9  Missing value

`scVel` (4-byte float, array size: XYZ x nscan):

The velocity vector (m s\(^{-1}\)) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9  Missing value

`scLat` (4-byte float, array size: nscan):

The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9  Missing value

`scLon` (4-byte float, array size: nscan):

The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value
scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:  
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:  
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:  
-9999.9 Missing value

scAttPitchGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:  
-9999.9 Missing value

scAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:  
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
  -9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
  -9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
  -9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
  -9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
  -9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
  -9999.9 Missing value

### Input (Group in NS)

**surfaceElevation** (4-byte float, array size: nrayNS x nscan):
Altitudes above the earth ellipsoid of the surface gates from 2AKu. Values are in m. Special values are defined as:
  -9999.9 Missing value

**surfaceType** (4-byte integer, array size: nrayNS x nscan):
Surface type from 2AKu. Special values are defined as:
  -9999 Missing value

**localZenithAngle** (4-byte float, array size: nrayNS x nscan):
Zenith angle of the ray at the earth’s surface from 2AKu. Values are in degree. Special values are defined as:
  -9999.9 Missing value
**precipitationFlag** (4-byte integer, array size: nrayNS x nscan):
Precipitation flag from 2AKu. Special values are defined as:
-9999 Missing value

**surfaceRangeBin** (2-byte integer, array size: nrayNS x nscan):
Index of the surface range bin from 2AKu. Special values are defined as:
-9999 Missing value

**lowestClutterFreeBin** (2-byte integer, array size: nrayNS x nscan):
Index of lowest clutter-free bin from 2AKu. Special values are defined as:
-9999 Missing value

**ellipsoidBinOffset** (4-byte float, array size: nrayNS x nscan):
Offset of surface bin from the earth ellipsoid from 2AKu. Values are in m. Special values are defined as:
-9999.9 Missing value

**stormTopBin** (2-byte integer, array size: nrayNS x nscan):
Index of storm top bin from 2AKu. Special values are defined as:
-9999 Missing value

**stormTopAltitude** (4-byte float, array size: nrayNS x nscan):
Altitude of storm top bin from 2AKu. Values are in m. Special values are defined as:
-9999.9 Missing value

**zeroDegBin** (2-byte integer, array size: nrayNS x nscan):
Range bin of the freezing level. Special values are defined as:
-9999 Missing value

**zeroDegAltitude** (4-byte float, array size: nrayNS x nscan):
Altitude of the freezing level. Values are in m. Special values are defined as:
-9999.9 Missing value

**precipitationType** (4-byte integer, array size: nrayNS x nscan):
Precipitation type classification from 2AKu. Special values are defined as:
-9999 Missing value

**precipTypeQualityFlag** (4-byte integer, array size: nrayNS x nscan):
Quality flag of precipitation type from 2AKu. Special values are defined as:
-9999 Missing value

**piaEffective** (4-byte float, array size: nrayNS x nscan):
Effective 2-way PIA from 2AKu. Values are in dB. Special values are defined as:
-9999.9 Missing value

**piaEffectiveSigma** (4-byte float, array size: nrayNS x nscan):
Effective PIA uncertainty from 2AKu. Values are in dB. Special values are defined as:
-9999.9 Missing value

**piaEffectiveReliabFlag** (2-byte integer, array size: nrayNS x nscan):
Reliability flag of effective PIA from 2AKu. Special values are defined as:
-9999 Missing value
sigmaZeroMeasured (4-byte float, array size: nrayNS x nscan):
The surface normalized radar cross section. Values range from -40 to 42 dB. Special values are defined as:
-9999.9 Missing value

snowIceCover (4-byte integer, array size: nrayNS x nscan):
Snow and ice cover. Values are defined as: 0 = ice-free ocean 1 = snow-free land 2 = snow-covered land 3 = sea ice. Special values are defined as:
-9999 Missing value

aPriori (Group in NS)

profClass (4-byte integer, array size: nrayNS x nscan):
The class number of the observed reflectivity profile using a classification based upon measured reflectivity structure features. Unclassified profiles are assigned a value of -9999.

prinComp (4-byte float, array size: ncomp x nrayNS x nscan):
Principal components of the observed reflectivity profile, up to ncomp in number, that describe the primary modes of reflectivity structural variability. Unused principal components are assigned a value of -9999.9.

surfPrecipBiasRatio (4-byte float, array size: nrayNS x nscan):
The a priori ratio of mean MS-mode to NS-mode surface rain rates for the given observed reflectivity profile. Special values are defined as:
-9999.9 Missing value

initNw (4-byte float, array size: nBnPSDlo x nrayNS x nscan):
The initial values of the ensemble-mean, low-resolution (nBnPSDlo bins) profile of Nw associated with a given observed reflectivity profile. Nw is the intercept of the normalized gamma distribution used to describe the precipitation particle size distribution. Special values are defined as:
-9999.9 Missing value

surfaceAirPressure (4-byte float, array size: nrayNS x nscan):
Surface air pressure. Values range from 300 to 1100 hPa. Special values are defined as:
-9999.9 Missing value

surfaceAirTemperature (4-byte float, array size: nrayNS x nscan):
Surface air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

surfaceVaporDensity (4-byte float, array size: nrayNS x nscan):
Surface vapor density. Values range from 0 to 60 g/m$^3$. Special values are defined as:
-9999.9 Missing value
skinTemperature (4-byte float, array size: nrayNS x nscan):
Surface skin temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

envParamNode (2-byte integer, array size: nBnEnv x nrayNS x nscan):
Bin indices for environmental parameters. Special values are defined as:
-9999 Missing value

airPressure (4-byte float, array size: nBnEnv x nrayNS x nscan):
Air pressure. Values range from 50 to 1100 hPa. Special values are defined as:
-9999.9 Missing value

airTemperature (4-byte float, array size: nBnEnv x nrayNS x nscan):
Air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

vaporDensity (4-byte float, array size: nBnEnv x nrayNS x nscan):
Vapor density. Values range from 0 to 60 $g/m^3$. Special values are defined as:
-9999.9 Missing value

cloudLiqWaterCont (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Cloud liquid water content. Values range from 0 to 60 $g/m^3$. Special values are defined as:
-9999.9 Missing value

cloudIceWaterCont (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Cloud ice water content. Values range from 0 to 18 $g/m^3$. Special values are defined as:
-9999.9 Missing value

phaseBinNodes (2-byte integer, array size: nPhsBnN x nrayNS x nscan):
Bin numbers indicating (0) storm top, (1) top of mixed-phase layer, (2) maximum reflectivity in mixed-phase layer if bright band detected; otherwise, the freezing level from analysis, (3) bottom of mixed-phase layer, and (4) bottom of rain layer Special values are defined as:
-9999 Missing value

PSDparamLowNode (2-byte integer, array size: nBnPSDlo x nrayNS x nscan):
Bin indices for low-resolution PSD parameters. Special values are defined as:
-9999 Missing value

precipTotPSDparamLow (4-byte float, array size: nPSDlo x nBnPSDlo x nrayNS x nscan):
Total precipitation low-resolution PSD parameters. Parameters are log10(Nw) with units log10(1 / m^4) for first value of nPSDlo, mu with no units for second value.

precipTotPSDparamHigh (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation high-resolution PSD parameters. Values range from 0 to 20 mm_Dm. Special values are defined as:
-9999.9 Missing value
**precipTotWaterCont** (4-byte float, array size: nBnPsdhi x nRayNS x nScan):
Total precipitation liquid water content. Values range from 0 to 18 g/m$^3$. Special values are defined as:
- 9999.9 Missing value

**precipTotWaterContSigma** (4-byte float, array size: nBnPsdhi x nRayNS x nScan):
Total precipitation liquid water content uncertainty. Values range from 0 to 18 g/m$^3$. Special values are defined as:
- 9999.9 Missing value

**precipTotRate** (4-byte float, array size: nBnPsdhi x nRayNS x nScan):
Total precipitation rate. Values range from 0 to 300 mm/hr. Special values are defined as:
- 99 No precipitation detected
- 9999.9 Missing value

**precipTotRateSigma** (4-byte float, array size: nBnPsdhi x nRayNS x nScan):
Total precipitation rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
- 99 No precipitation detected
- 9999.9 Missing value

**liqMassFracTrans** (4-byte float, array size: nBnTr x nRayNS x nScan):
Fraction of the precipitation mass that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
- 9999.9 Missing value

**liqRateFracTrans** (4-byte float, array size: nBnTr x nRayNS x nScan):
Fraction of the precipitation rate that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
- 9999.9 Missing value

**surfPrecipTotRate** (4-byte float, array size: nRayNS x nScan):
Surface rain rate. Values range from 0 to 300 mm/hr. Special values are defined as:
- 99 No precipitation detected
- 9999.9 Missing value

**surfPrecipTotRateSigma** (4-byte float, array size: nRayNS x nScan):
Surface rain rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
- 99 No precipitation detected
- 9999.9 Missing value

**surfLiqRateFrac** (4-byte float, array size: nRayNS x nScan):
Surface liquid precipitation rate fraction. Values range from 0 to 1. Special values are
defined as:
  -9999.9  Missing value

**tenMeterWindSpeed** (4-byte float, array size: nrayNS x nscan):
Ten meter altitude wind speed magnitude. Values range from 0 to 100 m/s. Special values are defined as:
  -9999.9  Missing value

**surfEmissivity** (4-byte float, array size: nemiss x nrayNS x nscan):
GMI emissivities. Values range from 0 to 1. Special values are defined as:
  -9999.9  Missing value

**simulatedBrightTemp** (4-byte float, array size: nemiss x nrayNS x nscan):
GMI simulated brightness temperatures. Values range from 20 to 350 K. Special values are defined as:
  -9999.9  Missing value

**nubfPIAfactor** (4-byte float, array size: nrayNS x nscan):
nubfPIAfactor is the factor applied to the Hitschfeld-Bordan path integrated attenuation to obtain the simulated path integrated attenuation, accounting for the nonuniform beamfilling by precipitation which is estimated from a 3x3 neighborhood of footprints. Values range from 20 to 350. Special values are defined as:
  -9999.9  Missing value

**multiScatMaxContrib** (4-byte float, array size: nrayNS x nscan):
multiScatMaxContrib is the maximum contribution, in a given radar profile, by multiple scattering to the simulated reflectivity. Values range from 20 to 350 dB. Special values are defined as:
  -9999.9  Missing value

**surfEmissSigma** (4-byte float, array size: nemiss x nrayNS x nscan):
Values range from 20 to 350. Special values are defined as:
  -9999.9  Missing value

**tenMeterWindSigma** (4-byte float, array size: nrayNS x nscan):
Values range from 0 to 100 m/s. Special values are defined as:
  -9999.9  Missing value

**skinTempSigma** (4-byte float, array size: nrayNS x nscan):
Values range from 20 to 350 K. Special values are defined as:
  -9999.9  Missing value

**columnVaporSigma** (4-byte float, array size: nrayNS x nscan):
Values range from 20 to 350 kg/m$^2$. Special values are defined as:
  -9999.9  Missing value

**columnCloudLiqSigma** (4-byte float, array size: nrayNS x nscan):
Values range from 20 to 350 kg/m$^2$. Special values are defined as:
  -9999.9  Missing value
**errorOfDataFit** (4-byte float, array size: nrayNS x nscan):
Values range from 20 to 350 K. Special values are defined as:
-9999.9 Missing value

**pia** (4-byte float, array size: nrayNS x nscan):
Two-way path-integrated attenuation at Ku. Values range from 0 to 1000 dB. Special values are defined as:
-9999.9 Missing value

**correctedReflectFactor** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Corrected radar reflectivities at Ku band. Values range from -20 to 100 dBZ. Special values are defined as:
-9999.9 Missing value

**FLG** (Group in NS)

**ioQuality** (4-byte integer, array size: nrayNS x nscan):
Quality flag for input and output. The flag is a six digit number as follows.

1’s place
- 0 : rain estimate is valid
- 9 : no estimate (bad scan)

10’s place
- 0 : Ku data OK and rain detected using Ku
- 1 : Ku data OK and no rain detected using Ku
- 9 : bad Ku input data

100’s place
- 0 : Ku-SRT gives a valid PIA estimate
- 1 : sigma-zero at Ku is within the noise of the background
- 2 : sigma-zero at Ku is completely attenuated
- 9 : bad Ku input data

1000’s place
- 0 : freezing level is derived from Ku bright band
- 1 : freezing level is derived from GANAL analysis
- 9 : bad Ku input data

10000’s place
- 0 : Ku classified as stratiform or convective
- 1 : Ku classified as indeterminate
- 2 : precipitation not detected at Ku (no feature)
- 9 : bad Ku input data

100000’s place
- 0 : some measured Tb’s (interpolated to DPR grid) are valid
- 9 : no measured Tb’s are valid
Special values are defined as:
-9999 Missing value

**multiScatCalc** (4-byte integer, array size: nrayNS x nscan):

Special values are defined as:
-9999 Missing value

**algoType** (4-byte integer, array size: nrayNS x nscan):

Special values are defined as:
-9999 Missing value

**MS** (Swath)

**MS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in MS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nrayMS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nrayMS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in MS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>
**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in
dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be
useful. A zero integer value indicates usual geolocation. A non-zero value broken down
into the following bit flags indicates the following, where bit 0 is the least significant bit
(i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
</tbody>
</table>
Anomalous Time Step
GHA not calculated due to error
SunData (Group) not calculated due to error
Failure to calculate Sun in inertial coordinates
Fallback to GES ephemeris
Fallback to GEONS ephemeris
Fallback to PVT ephemeris
Fallback to OBP ephemeris
Spare (always 0)
Spare (always 0)
Spare (always 0)
Spare (always 0)

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
</tbody>
</table>
2  SUNPOINT
3  GSPM (Gyro-less Sun Point)
4  MSM (Mission Science Mode)
5  SLEW
6  DELTAH
7  DELTAV
-99  UNKNOWN -- ACS mode unavailable

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
</tbody>
</table>
17  Ku/Ka Independent Standby VPRF Table OUT
18  Ku/Ka Independent Standby Phase Out
19  Ku/Ka Independent Standby Dump Out
20  Ku/Ka Independent Standby (No Science Data)

**limitErrorFlag** (1-byte integer, array size: nscan):

- Bit flags for every ray with information about echo power limit checks.
- limitErrorFlag may be used in modeStatus.
- Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
- 9999.9  Missing value

**navigation** (Group in MS)

**scPos** (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9  Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector ($ms^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9  Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9  Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value
**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
5 STANDARD GPM PRODUCTS

are defined as:
-9999.9  Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9  Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9  Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 1000000000 s. Special values are defined as:
-9999.9  Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9  Missing value

**Input** (Group in MS)

**surfaceElevation** (4-byte float, array size: nrayMS x nscan):
Altitudes above the earth ellipsoid of the surface gates from 2ADPR. Values are in m. Special values are defined as:
-9999.9  Missing value

**surfaceType** (4-byte integer, array size: nrayMS x nscan):
Surface type from 2ADPR. Special values are defined as:
-9999  Missing value

**localZenithAngle** (4-byte float, array size: nrayMS x nscan):
Zenith angle of the ray at the earth’s surface from 2ADPR. Values are in degree. Special values are defined as:
-9999.9  Missing value
**Precipitation Flag** (4-byte integer, array size: nKuKa x nrayMS x nscan):
Precipitation flag from 2ADPR (Ku/Ka). Special values are defined as:
-9999 Missing value

**Surface Range Bin** (2-byte integer, array size: nKuKa x nrayMS x nscan):
Index of the surface range bin from 2ADPR (Ku/Ka). Special values are defined as:
-9999 Missing value

**Lowest Clutter Free Bin** (2-byte integer, array size: nKuKa x nrayMS x nscan):
Index of lowest clutter-free bin from 2ADPR (Ku/Ka). Special values are defined as:
-9999 Missing value

**Ellipsoid Bin Offset** (4-byte float, array size: nKuKa x nrayMS x nscan):
Offset of surface bin from the Earth ellipsoid from 2ADPR (Ku/Ka). Values are in m. Special values are defined as:
-9999.9 Missing value

**Storm Top Bin** (2-byte integer, array size: nKuKa x nrayMS x nscan):
Index of storm top bin from 2ADPR (Ku/Ka). Special values are defined as:
-9999 Missing value

**Storm Top Altitude** (4-byte float, array size: nKuKa x nrayMS x nscan):
Altitude of storm top bin from 2ADPR (Ku/Ka). Values are in m. Special values are defined as:
-9999.9 Missing value

**Zero Deg Bin** (2-byte integer, array size: nKuKa x nrayMS x nscan):
Range bin of the freezing level. Special values are defined as:
-9999 Missing value

**Zero Deg Altitude** (4-byte float, array size: nKuKa x nrayMS x nscan):
Altitude of the freezing level. Values are in m. Special values are defined as:
-9999.9 Missing value

**Precipitation Type** (4-byte integer, array size: nKuKa x nrayMS x nscan):
Precipitation type classification from 2ADPR. Special values are defined as:
-9999 Missing value

**Precip Type Quality Flag** (4-byte integer, array size: nKuKa x nrayMS x nscan):
Quality flag of precipitation type from 2ADPR. Special values are defined as:
-9999 Missing value

**Pia Effective** (4-byte float, array size: nKuKa x nrayMS x nscan):
Effective 2-way PIA at Ku band from 2ADPR (Ku/Ka). Values are in dB. Special values are defined as:
-9999.9 Missing value

**Pia Effective Sigma** (4-byte float, array size: nKuKa x nrayMS x nscan):
Effective PIA uncertainty at Ku band from 2ADPR (Ku/Ka). Values are in dB. Special values are defined as:
-9999.9 Missing value
**piaEffectiveReliableFlag** (2-byte integer, array size: nKuKa x nrayMS x nscan):
Reliability flag of effective PIA from 2ADPR (Ku/Ka). Special values are defined as:
-9999 Missing value

**sigmaZeroMeasured** (4-byte float, array size: nrayMS x nscan):
The surface normalized radar cross section. Values range from -40 to 42 dB. Special values are defined as:
-9999.9 Missing value

**snowIceCover** (4-byte integer, array size: nrayMS x nscan):
Snow and ice cover. Values are defined as: 0 = ice-free ocean 1 = snow-free land 2 = snow-covered land 3 = sea ice. Special values are defined as:
-9999 Missing value

**aPriori** (Group in MS)

**profClass** (4-byte integer, array size: nrayMS x nscan):
The class number of the observed reflectivity profile using a classification based upon measured reflectivity structure features. Unclassified profiles are assigned a value of -9999.

**prinComp** (4-byte float, array size: ncomp x nrayMS x nscan):
Principal components of the observed reflectivity profile, up to ncomp in number, that describe the primary modes of reflectivity structural variability. Unused principal components are assigned a value of -9999.9.

**surfPrecipBiasRatio** (4-byte float, array size: nrayMS x nscan):
The a priori ratio of mean MS-mode to NS-mode surface rain rates for the given observed reflectivity profile. Special values are defined as:
-9999.9 Missing value

**initNw** (4-byte float, array size: nBnPsdLo x nrayMS x nscan):
The initial values of the ensemble-mean, low-resolution (nBnPsdLo bins) profile of Nw associated with a given observed reflectivity profile. Nw is the intercept of the normalized gamma distribution used to describe the precipitation particle size distribution. Special values are defined as:
-9999.9 Missing value

**surfaceAirTemperature** (4-byte float, array size: nrayMS x nscan):
Surface air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value
**surfaceVaporDensity** (4-byte float, array size: nrayMS x nscan):
Surface vapor density. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value

**skinTemperature** (4-byte float, array size: nrayMS x nscan):
Surface skin temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

**envParamNode** (2-byte integer, array size: nBnEnv x nrayMS x nscan):
Bin indices for environmental parameters. Special values are defined as:
-9999 Missing value

**airPressure** (4-byte float, array size: nBnEnv x nrayMS x nscan):
Air pressure. Values range from 50 to 1100 hPa. Special values are defined as:
-9999.9 Missing value

**airTemperature** (4-byte float, array size: nBnEnv x nrayMS x nscan):
Air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

**vaporDensity** (4-byte float, array size: nBnEnv x nrayMS x nscan):
Vapor density. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value

**cloudLiqWaterCont** (4-byte float, array size: nBnPSDhi x nrayMS x nscan):
Cloud liquid water content. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value

**cloudIceWaterCont** (4-byte float, array size: nBnPSDhi x nrayMS x nscan):
Cloud ice water content. Values range from 0 to 18 g/m³. Special values are defined as:
-9999.9 Missing value

**phaseBinNodes** (2-byte integer, array size: nPhsBnN x nrayMS x nscan):
Bin numbers indicating (0) storm top, (1) top of mixed-phase layer, (2) maximum reflectivity in mixed-phase layer if bright band detected; otherwise, the freezing level from analysis, (3) bottom of mixed-phase layer, and (4) bottom of rain layer. Special values are defined as:
-9999 Missing value

**PSDparamLowNode** (2-byte integer, array size: nBnPSDlo x nrayMS x nscan):
Bin indices for low-resolution PSD parameters. Special values are defined as:
-9999 Missing value

**precipTotPSDparamLow** (4-byte float, array size: nPSDlo x nBnPSDlo x nrayMS x nscan):
Total precipitation low-resolution PSD parameters. Parameters are log10(Nw) with units log10(1 / m⁴) for first value of nPSDlo, mu with no units for second value.

**precipTotPSDparamHigh** (4-byte float, array size: nBnPSDhi x nrayMS x nscan):
Total precipitation high-resolution PSD parameters. Values range from 0 to 20 mm_Dm.
Special values are defined as:
-9999.9 Missing value

**precipTotWaterCont** (4-byte float, array size: nBnPSDhi x nrayMS x nscan):
Total precipitation liquid water content. Values range from 0 to 18 g/m$^3$. Special values are defined as:
-9999.9 Missing value

**precipTotWaterContSigma** (4-byte float, array size: nBnPSDhi x nrayMS x nscan):
Total precipitation liquid water content uncertainty. Values range from 0 to 18 g/m$^3$. Special values are defined as:
-9999.9 Missing value

**precipTotRate** (4-byte float, array size: nBnPSDhi x nrayMS x nscan):
Total precipitation rate. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

**precipTotRateSigma** (4-byte float, array size: nBnPSDhi x nrayMS x nscan):
Total precipitation rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

**liqMassFracTrans** (4-byte float, array size: nBnTr x nrayMS x nscan):
Fraction of the precipitation mass that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

**liqRateFracTrans** (4-byte float, array size: nBnTr x nrayMS x nscan):
Fraction of the precipitation rate that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

**surfPrecipTotRate** (4-byte float, array size: nrayMS x nscan):
Surface rain rate. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

**surfPrecipTotRateSigma** (4-byte float, array size: nrayMS x nscan):
Surface rain rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value
surfLiqRateFrac (4-byte float, array size: nrayMS x nscan):
Surface liquid precipitation rate fraction. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

tenMeterWindSpeed (4-byte float, array size: nrayMS x nscan):
Ten meter altitude wind speed magnitude. Values range from 0 to 100 m/s. Special values are defined as:
-9999.9 Missing value

surfEmissivity (4-byte float, array size: nemiss x nrayMS x nscan):
GMI emissivities. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

simulatedBrightTemp (4-byte float, array size: nemiss x nrayMS x nscan):
GMI simulated brightness temperatures. Values range from 20 to 350 K. Special values are defined as:
-9999.9 Missing value

nubfPIAfactor (4-byte float, array size: nrayMS x nscan):
nubfPIAfactor is the factor applied to the Hitschfeld-Bordan path integrated attenuation to obtain the simulated path integrated attenuation, accounting for the nonuniform beamfilling by precipitation which is estimated from a 3x3 neighborhood of footprints. Values range from 20 to 350. Special values are defined as:
-9999.9 Missing value

multiScatMaxContrib (4-byte float, array size: nrayMS x nscan):
multiScatMaxContrib is the maximum contribution, in a given radar profile, by multiple scattering to the simulated reflectivity. Values range from 20 to 350 dB. Special values are defined as:
-9999.9 Missing value

surfEmissSigma (4-byte float, array size: nemiss x nrayMS x nscan):
Values range from 20 to 350. Special values are defined as:
-9999.9 Missing value

tenMeterWindSigma (4-byte float, array size: nrayMS x nscan):
Values range from 0 to 100 m/s. Special values are defined as:
-9999.9 Missing value

skinTempSigma (4-byte float, array size: nrayMS x nscan):
Values range from 20 to 350 K. Special values are defined as:
-9999.9 Missing value

columnVaporSigma (4-byte float, array size: nrayMS x nscan):
Values range from 20 to 350 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

columnCloudLiqSigma (4-byte float, array size: nrayMS x nscan):
Values range from 20 to 350 kg/m$^2$. Special values are defined as:
-9999.9 Missing value

**errorOfDataFit** (4-byte float, array size: nrayMS x nscan):  
Values range from 20 to 350 K. Special values are defined as:  
-9999.9 Missing value

**pia** (4-byte float, array size: nKuKa x nrayMS x nscan):  
Two-way path-integrated attenuation (Ku/Ka). Values range from 0 to 1000 dB. Special values are defined as:  
-9999.9 Missing value

**correctedReflectFactor** (4-byte float, array size: nKuKa x nBnPSDhi x nrayMS x nscan):  
Corrected radar reflectivities (Ku/Ka). Values range from -20 to 100 dBZ. Special values are defined as:  
-9999.9 Missing value

**FLG** (Group in MS)

**ioQuality** (4-byte integer, array size: nrayMS x nscan):  
Quality flag for input and output. The flag is a six digit number as follows.

1’s place  
0 : rain estimate is valid  
9 : no estimate (bad scan)

10’s place  
0 : Ku data OK and rain detected using Ku  
1 : Ku data OK and no rain detected using Ku  
9 : bad Ku input data

100’s place  
0 : Ku-SRT gives a valid PIA estimate  
1 : sigma-zero at Ku is within the noise of the background  
2 : sigma-zero at Ku is completely attenuated  
9 : bad Ku input data

1000’s place  
0 : freezing level is derived from Ku bright band  
1 : freezing level is derived from GANAL analysis  
9 : bad Ku input data

10000’s place  
0 : Ku classified as stratiform or convective  
1 : Ku classified as indeterminate  
2 : precipitation not detected at Ku (no feature)  
9 : bad Ku input data

100000’s place  
0 : some measured Tb’s (interpolated to DPR grid)
are valid
9 : no measured Tb’s are valid

Special values are defined as:
-9999 Missing value

multiScatCalc (4-byte integer, array size: nrayMS x nscan):

Special values are defined as:
-9999 Missing value

algoType (4-byte integer, array size: nrayMS x nscan):

Special values are defined as:
-9999 Missing value

C Structure Header file:

```c
#ifndef _TK_2BCMB_H_
#define _TK_2BCMB_H_

#ifndef _L2BCMB_MS_FLG_
#define _L2BCMB_MS_FLG_

typedef struct {
    int ioQuality[25];
    int multiScatCalc[25];
    int algoType[25];
} L2BCMB_MS_FLG;
#endif

#ifndef _L2BCMB_MS_APRIORI_
#define _L2BCMB_MS_APRIORI_

typedef struct {
    int profClass[25];
    float prinComp[25][5];
    float surfPrecipBiasRatio[25];
#endif
```

float initNw[25][9];
} L2BCMB_MS_APRIORI;

#endif

#ifndef _L2BCMB_MS_INPUT_
define _L2BCMB_MS_INPUT_

typedef struct {
    float surfaceElevation[25];
    int surfaceType[25];
    float localZenithAngle[25];
    int precipitationFlag[25][2];
    short surfaceRangeBin[25][2];
    short lowestClutterFreeBin[25][2];
    float ellipsoidBinOffset[25][2];
    short stormTopBin[25][2];
    float stormTopAltitude[25][2];
    short zeroDegBin[25][2];
    float zeroDegAltitude[25];
    int precipitationType[25];
    int precipTypeQualityFlag[25];
    float piaEffective[25][2];
    float piaEffectiveSigma[25][2];
    short piaEffectiveReliabFlag[25][2];
    float sigmaZeroMeasured[25];
    int snowIceCover[25];
} L2BCMB_MS_INPUT;
#endif

#ifndef _L2BCMB_MS_SCANSTATUS_
define _L2BCMB_MS_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
}
signed char acsModeMidScan;
signed char targetSelectionMidScan;
signed char operationalMode;
signed char limitErrorFlag;
double FractionalGranuleNumber;
} L2BCMB_MS_SCANSTATUS;

#endif

#ifndef _L2BCMB_MS_
#define _L2BCMB_MS_

typedef struct {
  SCANTIME ScanTime;
  float Latitude[25];
  float Longitude[25];
  L2BCMB_MS_SCANSTATUS scanStatus;
  NAVIGATION navigation;
  L2BCMB_MS_INPUT Input;
  L2BCMB_MS_APRIORI aPriori;
  float surfaceAirPressure[25];
  float surfaceAirTemperature[25];
  float surfaceVaporDensity[25];
  float skinTemperature[25];
  short envParamNode[25][10];
  float airPressure[25][10];
  float airTemperature[25][10];
  float vaporDensity[25][10];
  float cloudLiqWaterCont[25][88];
  float cloudIceWaterCont[25][88];
  short phaseBinNodes[25][5];
  short PSDparamLowNode[25][9];
  float precipTotPSDparamLow[25][9][2];
  float precipTotPSDparamHigh[25][88];
  float precipTotWaterCont[25][88];
  float precipTotWaterContSigma[25][88];
  float precipTotRate[25][88];
  float precipTotRateSigma[25][88];
  float liqMassFracTrans[25][10];
  float liqRateFracTrans[25][10];
  float surfPrecipTotRate[25];
  float surfPrecipTotRateSigma[25];
  float surfLiqRateFrac[25];
} L2BCMB_MS_DATA;
float tenMeterWindSpeed[25];
float surfEmissivity[25][13];
float simulatedBrightTemp[25][13];
float nubfPIAfactor[25];
float multiScatMaxContrib[25];
float surfEmissSigma[25][13];
float tenMeterWindSigma[25];
float skinTempSigma[25];
float columnVaporSigma[25];
float columnCloudLiqSigma[25];
float errorOfDataFit[25];
float pia[25][2];
float correctedReflectFactor[25][88][2];
L2BCMB_MS_FLG FLG;
} L2BCMB_MS;

#endif

#ifndef _L2BCMB_NS_FLG_
#define _L2BCMB_NS_FLG_

typedef struct {
    int ioQuality[49];
    int multiScatCalc[49];
    int algoType[49];
} L2BCMB_NS_FLG;
#endif

#ifndef _L2BCMB_NS_APRIORI_
#define _L2BCMB_NS_APRIORI_

typedef struct {
    int profClass[49];
    float prinComp[49][5];
    float surfPrecipBiasRatio[49];
    float initNw[49][9];
} L2BCMB_NS_APRIORI;
#endif

#ifndef _L2BCMB_NS_INPUT_
#define _L2BCMB_NS_INPUT_

} L2BCMB_NS;

#endif

#ifndef _L2BCMB_NS_INPUT_
#define _L2BCMB_NS_INPUT_

} L2BCMB_NS;
typedef struct {
    float surfaceElevation[49];
    int surfaceType[49];
    float localZenithAngle[49];
    int precipitationFlag[49];
    short surfaceRangeBin[49];
    short lowestClutterFreeBin[49];
    float ellipsoidBinOffset[49];
    short stormTopBin[49];
    float stormTopAltitude[49];
    short zeroDegBin[49];
    float zeroDegAltitude[49];
    int precipitationType[49];
    int precipTypeQualityFlag[49];
    float piaEffective[49];
    float piaEffectiveSigma[49];
    short piaEffectiveReliabFlag[49];
    float sigmaZeroMeasured[49];
    int snowIceCover[49];
} L2BCMB_NS_INPUT;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} _NAVIGATION_;
typedef struct {
  signed char dataQuality;
  signed char dataWarning;
  signed char missing;
  signed char modeStatus;
  short geoError;
  short geoWarning;
  short SCorientation;
  short pointingStatus;
  signed char acsModeMidScan;
  signed char targetSelectionMidScan;
  signed char operationalMode;
  signed char limitErrorFlag;
  double FractionalGranuleNumber;
} L2BCMB_NS_SCANSTATUS;

typedef struct {
  short Year;
  signed char Month;
  signed char DayOfMonth;
  signed char Hour;
  signed char Minute;
  signed char Second;
  short MilliSecond;
  short DayOfYear;
  double SecondOfDay;
} SCANTIME;
```c
#define _L2BCMB_NS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2BCMB_NS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2BCMB_NS_INPUT Input;
    L2BCMB_NS_APIORI aPriori;
    float surfaceAirPressure[49];
    float surfaceAirTemperature[49];
    float surfaceVaporDensity[49];
    float skinTemperature[49];
    short envParamNode[49][10];
    float airPressure[49][10];
    float airTemperature[49][10];
    float vaporDensity[49][10];
    float cloudLiqWaterCont[49][88];
    float cloudIceWaterCont[49][88];
    short phaseBinNodes[49][5];
    short PSDparamLowNode[49][9];
    float precipTotPSDparamLow[49][9][2];
    float precipTotPSDparamHigh[49][88];
    float precipTotWaterCont[49][88];
    float precipTotWaterContSigma[49][88];
    float precipTotRate[49][88];
    float precipTotRateSigma[49][88];
    float liqMassFracTrans[49][10];
    float liqRateFracTrans[49][10];
    float surfPrecipTotRate[49];
    float surfPrecipTotRateSigma[49];
    float surfLiqRateFrac[49];
    float tenMeterWindSpeed[49];
    float surfEmissivity[49][13];
    float simulatedBrightTemp[49][13];
    float nubfPIAfactor[49];
    float multiScatMaxContrib[49];
    float surfEmissSigma[49][13];
    float tenMeterWindSigma[49];
    float skinTempSigma[49];
    float columnVaporSigma[49];
    float columnCloudLiqSigma[49];
};
```
float errorOfDataFit[49];
float pia[49];
float correctedReflectFactor[49][88];
L2BCMB_NS_FLG FLG;
} L2BCMB_NS;

#endif

#ifndef _L2BCMB_SWATHS_
#define _L2BCMB_SWATHS_

typedef struct {
    L2BCMB_NS NS;
    L2BCMB_MS MS;
} L2BCMB_SWATHS;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L2BCMB_MS_FLG/
    INTEGER*4 ioQuality(25)
    INTEGER*4 multiScatCalc(25)
    INTEGER*4 algoType(25)
END STRUCTURE

STRUCTURE /L2BCMB_MS_APRIORI/
    INTEGER*4 profClass(25)
    REAL*4 prinComp(5,25)
    REAL*4 surfPrecipBiasRatio(25)
    REAL*4 initNw(9,25)
END STRUCTURE

STRUCTURE /L2BCMB_MS_INPUT/
    REAL*4 surfaceElevation(25)
    INTEGER*4 surfaceType(25)
    REAL*4 localZenithAngle(25)
    INTEGER*4 precipitationFlag(2,25)
    INTEGER*2 surfaceRangeBin(2,25)
    INTEGER*2 lowestClutterFreeBin(2,25)
    REAL*4 ellipsoidBinOffset(2,25)
INTEGER*2 stormTopBin(2,25)
REAL*4 stormTopAltitude(2,25)
INTEGER*2 zeroDegBin(2,25)
REAL*4 zeroDegAltitude(25)
INTEGER*4 precipitationType(25)
INTEGER*4 precipTypeQualityFlag(25)
REAL*4 piaEffective(2,25)
REAL*4 piaEffectiveSigma(2,25)
INTEGER*2 piaEffectiveReliabFlag(2,25)
REAL*4 sigmaZeroMeasured(25)
INTEGER*4 snowIceCover(25)

END STRUCTURE

STRUCTURE /L2BCMB_MS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /L2BCMB_MS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(25)
  REAL*4 Longitude(25)
  RECORD /L2BCMB_MS_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L2BCMB_MS_INPUT/ Input
  RECORD /L2BCMB_MS_APRIORI/ aPriori
  REAL*4 surfaceAirPressure(25)
  REAL*4 surfaceAirTemperature(25)
  REAL*4 surfaceVaporDensity(25)
  REAL*4 skinTemperature(25)
  INTEGER*2 envParamNode(10,25)
  REAL*4 airPressure(10,25)
REAL*4 airTemperature(10,25)
REAL*4 vaporDensity(10,25)
REAL*4 cloudLiqWaterCont(88,25)
REAL*4 cloudIceWaterCont(88,25)
INTEGER*2 phaseBinNodes(5,25)
INTEGER*2 PSDparamLowNode(9,25)
REAL*4 precipTotPSDparamLow(2,9,25)
REAL*4 precipTotPSDparamHigh(88,25)
REAL*4 precipTotWaterCont(88,25)
REAL*4 precipTotWaterContSigma(88,25)
REAL*4 precipTotRate(88,25)
REAL*4 precipTotRateSigma(88,25)
REAL*4 liqMassFracTrans(10,25)
REAL*4 liqRateFracTrans(10,25)
REAL*4 surfPrecipTotRate(25)
REAL*4 surfPrecipTotRateSigma(25)
REAL*4 surfLiqRateFrac(25)
REAL*4 tenMeterWindSpeed(25)
REAL*4 surfEmissivity(13,25)
REAL*4 simulatedBrightTemp(13,25)
REAL*4 nubfPIAfactor(25)
REAL*4 multiScatMaxContrib(25)
REAL*4 surfEmissSigma(13,25)
REAL*4 tenMeterWindSigma(25)
REAL*4 skinTempSigma(25)
REAL*4 columnVaporSigma(25)
REAL*4 columnCloudLiqSigma(25)
REAL*4 errorOfDataFit(25)
REAL*4 pia(2,25)
REAL*4 correctedReflectFactor(2,88,25)
RECORD /L2BCMB_MS_FLG/ FLG
END STRUCTURE

STRUCTURE /L2BCMB_NS_FLG/
   INTEGER*4 ioQuality(49)
   INTEGER*4 multiScatCalc(49)
   INTEGER*4 algoType(49)
END STRUCTURE

STRUCTURE /L2BCMB_NS_APRIORI/
   INTEGER*4 profClass(49)
   REAL*4 prinComp(5,49)
   REAL*4 surfPrecipBiasRatio(49)
REAL*4 initNw(9,49)
END STRUCTURE

STRUCTURE /L2BCMB_NS_INPUT/
  REAL*4 surfaceElevation(49)
  INTEGER*4 surfaceType(49)
  REAL*4 localZenithAngle(49)
  INTEGER*4 precipitationFlag(49)
  INTEGER*2 surfaceRangeBin(49)
  INTEGER*2 lowestClutterFreeBin(49)
  REAL*4 ellipsoidBinOffset(49)
  INTEGER*2 stormTopBin(49)
  REAL*4 stormTopAltitude(49)
  INTEGER*2 zeroDegBin(49)
  REAL*4 zeroDegAltitude(49)
  INTEGER*4 precipitationType(49)
  INTEGER*4 precipTypeQualityFlag(49)
  REAL*4 piaEffective(49)
  REAL*4 piaEffectiveSigma(49)
  INTEGER*2 piaEffectiveReliabFlag(49)
  REAL*4 sigmaZeroMeasured(49)
  INTEGER*4 snowIceCover(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
  REAL*4 scAlt
  REAL*4 dprAlt
  REAL*4 scAttRollGeoc
  REAL*4 scAttPitchGeoc
  REAL*4 scAttYawGeoc
  REAL*4 scAttRollGeod
  REAL*4 scAttPitchGeod
  REAL*4 scAttYawGeod
  REAL*4 greenHourAng
  REAL*8 timeMidScan
  REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L2BCMB_NS_SCANSTATUS/
BYTE dataQuality
BYTE dataWarning
BYTE missing
BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
BYTE limitErrorFlag
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /SCANTIME/

INTEGER*2 Year
BYTE Month
BYTE DayOfMonth
BYTE Hour
BYTE Minute
BYTE Second
INTEGER*2 MilliSecond
INTEGER*2 DayOfYear
REAL*8 SecondOfDay

END STRUCTURE

STRUCTURE /L2BCMB_NS/

RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(49)
REAL*4 Longitude(49)
RECORD /L2BCMB_NS_SCANSTATUS/ scanStatus
RECORD /NAVIGATION/ navigation
RECORD /L2BCMB_NS_INPUT/ Input
RECORD /L2BCMB_NS_APRIORI/ aPriori
REAL*4 surfaceAirPressure(49)
REAL*4 surfaceAirTemperature(49)
REAL*4 surfaceVaporDensity(49)
REAL*4 skinTemperature(49)
INTEGER*2 envParamNode(10,49)
REAL*4 airPressure(10,49)
REAL*4 airTemperature(10,49)
REAL*4 vaporDensity(10,49)
5.57 3CMB - Combined precipitation

3CMB, "Combined precipitation", computes statistics of the Combined measurements at both a low horizontal resolution (G1, 5° x 5° latitude/longitude) and a high horizontal resolution (G2, 0.25° x 0.25° latitude/longitude). There will be both a monthly product and a daily product.
Units and ranges not included in this version. When units and ranges are provided and no more changes are coming then they could be added. Use specific reference for each variable.

Dimension definitions:
- \( l_tL \) 28 Number of low resolution 5° grid intervals of latitude from 70°S to 70°N.
- \( l_nL \) 72 Number of low resolution 5° grid intervals of longitude from 180°W to 180°E.
- \( l_tH \) 536 Number of high resolution 0.25° grid intervals of latitude from 67°S to 67°N.
- \( l_nH \) 1440 Number of high resolution 0.25° grid intervals of longitude from 180°W to 180°E.
- \( n_s \) 2 Number of swaths: MS (Ku+Ka+microwave), NS (Ku+microwave).
- \( h_gt \) 16 Number of level heights 0-15: 0: near surface, 1-10: height = 1.0km * index, 11-15: height = 10.0km + 2.0km * (index-10),
- \( t_i_m \) 24 Number of hourly local time bins.
- \( r_t \) 3 Number of rain types: stratiform, convective, all.
- \( s_t \) 3 Number of surface types: ocean, land, all.
- \( b_i_n \) 30 Number of bins in histogram.

Figure 808 through Figure 826 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 809: Data Format Structure for 3CMB, G1
Figure 810: Data Format Structure for 3CMB, G2

Figure 811: Data Format Structure for 3CMB, G1, precipTotRate

Figure 812: Data Format Structure for 3CMB, G1, precipLiqRate
**5.57 3CMB - Combined precipitation**

- **precipTotWaterContent**
  - count: 4 bytes
  - mean: 4 bytes
  - stdev: 4 bytes
  - hist: 4 bytes
  - Array: ltL x lnL x ns x hgt x rt x st

- **precipLiqWaterContent**
  - count: 4 bytes
  - mean: 4 bytes
  - stdev: 4 bytes
  - hist: 4 bytes
  - Array: ltL x lnL x ns x hgt x rt x st

- **cloudLiqWaterContent**
  - count: 4 bytes
  - mean: 4 bytes
  - stdev: 4 bytes
  - hist: 4 bytes
  - Array: ltL x lnL x ns x hgt x st

- **precipTotDm**
  - count: 4 bytes
  - mean: 4 bytes
  - stdev: 4 bytes
  - hist: 4 bytes
  - Array: ltL x lnL x ns x hgt x rt x st

- **precipTotLogNw**
  - count: 4 bytes
  - mean: 4 bytes
  - stdev: 4 bytes
  - hist: 4 bytes
  - Array: ltL x lnL x ns x hgt x rt x st

- **surfPrecipTotRateDiurnal**
  - count: 4 bytes
  - mean: 4 bytes
  - stdev: 4 bytes
  - Array: ltL x lnL x ns x st x tim
Figure 819: Data Format Structure for 3CMB, G2, precipTotRate
5.57 3CMB - Combined precipitation

**FileHeader** (Metadata):  
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputFileNames** (Metadata):  
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

**InputAlgorithmVersions** (Metadata):  
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

**InputGenerationDateTime** (Metadata):  
InputGenerationDateTime contains a list of input generation datetimes. See Metadata for GPM Products for details.

**FileInfo** (Metadata):  
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**Grids** (Group)

Figure 820: Data Format Structure for 3CMB, G2, precipLiqRate

Figure 821: Data Format Structure for 3CMB, G2, precipTotWaterContent

Figure 822: Data Format Structure for 3CMB, G2, precipLiqWaterContent
Figure 823: Data Format Structure for 3CMB, G2, cloudLiqWaterContent

Figure 824: Data Format Structure for 3CMB, G2, precipTotDm

Figure 825: Data Format Structure for 3CMB, G2, precipTotLogNw

Figure 826: Data Format Structure for 3CMB, G2, surfPrecipTotRateDiurnal
G1 (Grid)

**G1_GridHeader** (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**precipTotRate** (Group in G1)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitation water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

- **count** (4-byte integer, array: ltL x lnL x ns x hgt x rt x st):
  Count. Special values are defined as:
  -9999  Missing value

- **mean** (4-byte float, array: ltL x lnL x ns x hgt x rt x st):
  mean. Special values are defined as:
  -9999.9  Missing value

- **stdev** (4-byte float, array: ltL x lnL x ns x hgt x rt x st):
  Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
  -9999.9  Missing value

- **hist** (4-byte integer, array: ltL x lnL x ns x hgt x rt x st x bin):
  Histogram. Special values are defined as:
  -9999  Missing value

**precipLiqRate** (Group in G1)
Equivalent precipitation rate of liquid-phase precipitating water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

- **count** (4-byte integer, array: ltL x lnL x ns x hgt x rt x st):
  Count. Special values are defined as:
  -9999  Missing value

- **mean** (4-byte float, array: ltL x lnL x ns x hgt x rt x st):
  mean. Special values are defined as:
  -9999.9  Missing value

- **stdev** (4-byte float, array: ltL x lnL x ns x hgt x rt x st):
  Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
  -9999.9  Missing value
hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999   Missing value

precipTotWaterContent (Group in G1)
Equivalent water content of both liquid-phase and ice-phase precipitating water (g/m$^3$).
(Note: liquid can be in the form of rain or melt water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999   Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999   Missing value

precipLiqWaterContent (Group in G1)
Equivalent water content of liquid-phase precipitating water (g/m$^3$). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999   Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999   Missing value
**cloudLiqWaterContent** (Group in G1)
Equivalent water content of liquid-phase cloud water \((g/m^3)\).

**count** (4-byte integer, array size: ltL x lnL x ns x hgt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x hgt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x hgt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x ns x hgt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipTotDm** (Group in G1)
Volume-weighted mean of the liquid-equivalent precipitation particle diameter (mm).

**count** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipTotLogNw** (Group in G1)
Common logarithm of the intercept of the normalized gamma distribution representing the liquid-equivalent precipitation particle size distribution \((\log_{10}(m^{-4}))\).
**count** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipAllObs** (4-byte integer, array size: ltL x lnL x ns x hgt x st):
Number of total observations, whether precipitating or not. Special values are defined as:
-9999 Missing value

**surfPrecipTotRateDiurnal** (Group in G1)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitating water in the lowest uncontaminated range-bin (mm/hr), indexed by the local time. (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltL x lnL x ns x st x tim):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x st x tim):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x st x tim):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**surfPrecipTotRateDiurnalAllObs** (4-byte integer, array size: ltL x lnL x ns x st x tim):
Number of total diurnal observations, whether precipitating or not. Special values are defined as:
-9999 Missing value
surfPrecipTotRateUn (4-byte float, array size: ltL x lnL x ns):
Surface total precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9  Missing value

surfPrecipLiqRateUn (4-byte float, array size: ltL x lnL x ns):
Surface liquid precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9  Missing value

surfPrecipTotRateProb (4-byte float, array size: ltL x lnL x ns):
Probability of total surface precipitation. Special values are defined as:
-9999.9  Missing value

surfPrecipLiqRateProb (4-byte float, array size: ltL x lnL x ns):
Probability of liquid surface precipitation. Special values are defined as:
-9999.9  Missing value

G2 (Grid)

G2_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipTotRate (Group in G2)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitation water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9  Missing value
precipLiqRate (Group in G2)
Equivalent precipitation rate of liquid-phase precipitating water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

precipTotWaterContent (Group in G2)
Equivalent water content of both liquid-phase and ice-phase precipitating water \((g/m^3)\). (Note: liquid can be in the form of rain or melt water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

precipLiqWaterContent (Group in G2)
Equivalent water content of liquid-phase precipitating water \((g/m^3)\). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value
**stddev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

**cloudLiqWaterContent** (Group in G2)
Equivalent water content of liquid-phase cloud water ($g/m^3$).

**count** (4-byte integer, array size: ltH x lnH x ns x hgt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt):
mean. Special values are defined as:
-9999.9 Missing value

**stddev** (4-byte float, array size: ltH x lnH x ns x hgt):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

**precipTotDm** (Group in G2)
Volume-weighted mean of the liquid-equivalent precipitation particle diameter (mm).

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stddev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

**precipTotLogNw** (Group in G2)
Common logarithm of the intercept of the normalized gamma distribution representing
the liquid-equivalent precipitation particle size distribution ($\log_{10}(m^{-4})$).

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value
**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipAllObs** (4-byte integer, array size: ltH x lnH x ns x hgt):
Number of total observations, whether precipitating or not. Special values are defined as:
-9999 Missing value

**surfPrecipTotRateDiurnal** (Group in G2)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitating water in the lowest uncontaminated range-bin (mm/hr), indexed by the local time. (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x tim):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x tim):
Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x tim):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**surfPrecipTotRateDiurnalAllObs** (4-byte integer, array size: ltH x lnH x ns x tim):
Number of total diurnal observations, whether precipitating or not. Special values are defined as:
-9999 Missing value

**surfPrecipTotRateUn** (4-byte float, array size: ltH x lnH x ns):
Surface total precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9 Missing value

**surfPrecipLiqRateUn** (4-byte float, array size: ltH x lnH x ns):
Surface liquid precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9 Missing value
surfPrecipTotRateProb (4-byte float, array size: ltH x lnH x ns):
Probability of total surface precipitation. Special values are defined as:
-9999.9 Missing value

surfPrecipLiqRateProb (4-byte float, array size: ltH x lnH x ns):
Probability of liquid surface precipitation. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_3CMB_H_
#define _TK_3CMB_H_

#ifndef _L3CMB_G2_SURFPRECIPTOTRATEDIURNAL_
#define _L3CMB_G2_SURFPRECIPTOTRATEDIURNAL_

typedef struct {
    int count[24][2][1440][536];
    float mean[24][2][1440][536];
    float stdev[24][2][1440][536];
} L3CMB_G2_SURFPRECIPTOTRATEDIURNAL;
#endif

#ifndef _L3CMB_G2_PRECIPTOTLOGNW_
#define _L3CMB_G2_PRECIPTOTLOGNW_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMB_G2_PRECIPTOTLOGNW;
#endif

#ifndef _L3CMB_G2_PRECIPTOTDM_
#define _L3CMB_G2_PRECIPTOTDM_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMB_G2_PRECIPTOTDM;
```
typedef struct {
    int count[16][2][1440][536];
    float mean[16][2][1440][536];
    float stdev[16][2][1440][536];
} L3CMB_G2_CLOUDLIQWATERCONTENT;

#endif

#ifndef _L3CMB_G2_PRECIPLIQWATERCONTENT_
#define _L3CMB_G2_PRECIPLIQWATERCONTENT_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMB_G2_PRECIPLIQWATERCONTENT;

#endif

#ifndef _L3CMB_G2_PRECIPTOTWATERCONTENT_
#define _L3CMB_G2_PRECIPTOTWATERCONTENT_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMB_G2_PRECIPTOTWATERCONTENT;

#endif

#ifndef _L3CMB_G2_PRECIPLIQRATE_
#define _L3CMB_G2_PRECIPLIQRATE_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMB_G2_PRECIPLIQRATE;
typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMB_G2_PRECIPTOTRATE;

typedef struct {
    L3CMB_G2_PRECIPTOTRATE precipTotRate;
    L3CMB_G2_PRECIPLIQRATE precipLiqRate;
    L3CMB_G2_PRECIPTOTWATERCONTENT precipTotWaterContent;
    L3CMB_G2_PRECIPLIQWATERCONTENT precipLiqWaterContent;
    L3CMB_G2_CLOUDLIQWATERCONTENT cloudLiqWaterContent;
    L3CMB_G2_PRECIPTOTDM precipTotDm;
    L3CMB_G2_PRECIPTOTLOGNW precipTotLogNw;
    int precipAllObs[16][2][1440][536];
    int surfPrecipTotRateDiurnalAllObs[24][2][1440][536];
    float surfPrecipTotRateUn[2][1440][536];
    float surfPrecipLiqRateUn[2][1440][536];
    float surfPrecipTotRateProb[2][1440][536];
    float surfPrecipLiqRateProb[2][1440][536];
} L3CMB_G2;

typedef struct {
    int count[24][3][2][72][28];
    float mean[24][3][2][72][28];
    float stdev[24][3][2][72][28];
typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stddev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMB_G1_PRECIPTOTLOGNW;

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stddev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMB_G1_PRECIPTOTDM;

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stddev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMB_G1_CLOUDLIQWATERCONTENT;

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stddev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMB_G1_PRECIPLIQWATERCONTENT;
typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMB_G1_PRECIPLIQWATERCONTENT;

#ifendif

#ifndef _L3CMB_G1_PRECIPTOTWATERCONTENT_
#define _L3CMB_G1_PRECIPTOTWATERCONTENT_

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMB_G1_PRECIPTOTWATERCONTENT;

#endif

#ifndef _L3CMB_G1_PRECIPLIQRATE_
#define _L3CMB_G1_PRECIPLIQRATE_

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMB_G1_PRECIPLIQRATE;

#endif

#ifndef _L3CMB_G1_PRECIPTOTRATE_
#define _L3CMB_G1_PRECIPTOTRATE_

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMB_G1_PRECIPTOTRATE;

#endif
typedef struct {
    L3CMB_G1_PRECIPTOTRATE precipTotRate;
    L3CMB_G1_PRECIPLIQRATE precipLiqRate;
    L3CMB_G1_PRECIPTOTWATERCONTENT precipTotWaterContent;
    L3CMB_G1_PRECIPLIQWATERCONTENT precipLiqWaterContent;
    L3CMB_G1_CLOUDLIQWATERCONTENT cloudLiqWaterContent;
    L3CMB_G1_PRECIPTOTDM precipTotDm;
    L3CMB_G1_PRECIPTOTLOGNW precipTotLogNw;
    int precipAllObs[3][16][2][72][28];
    L3CMB_G1_SURFPRECIPTOTRATEDIURNAL surfPrecipTotRateDiurnal;
    int surfPrecipTotRateDiurnalAllObs[24][3][2][72][28];
    float surfPrecipTotRateUn[2][72][28];
    float surfPrecipLiqRateUn[2][72][28];
    float surfPrecipTotRateProb[2][72][28];
    float surfPrecipLiqRateProb[2][72][28];
} L3CMB_G1;

typedef struct {
    L3CMB_G1 G1;
    L3CMB_G2 G2;
} L3CMB_GRIDS;

Fortran Structure Header file:

STRUCTURE /L3CMB_G2_SURFPRECIPTOTRATEDIURNAL/
    INTEGER*4 count(536,1440,2,24)
    REAL*4 mean(536,1440,2,24)
    REAL*4 stdev(536,1440,2,24)
END STRUCTURE
STRUCTURE /L3CMB_G2_PRECIPTOTLOGNW/
   INTEGER*4 count(536,1440,2,16,3)
   REAL*4 mean(536,1440,2,16,3)
   REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMB_G2_PRECIPTOTDM/
   INTEGER*4 count(536,1440,2,16,3)
   REAL*4 mean(536,1440,2,16,3)
   REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMB_G2_CLOUDLIQWATERCONTENT/
   INTEGER*4 count(536,1440,2,16)
   REAL*4 mean(536,1440,2,16)
   REAL*4 stdev(536,1440,2,16)
END STRUCTURE

STRUCTURE /L3CMB_G2_PRECIPLIQUWATERCONTENT/
   INTEGER*4 count(536,1440,2,16,3)
   REAL*4 mean(536,1440,2,16,3)
   REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMB_G2_PRECIPTOTWATERCONTENT/
   INTEGER*4 count(536,1440,2,16,3)
   REAL*4 mean(536,1440,2,16,3)
   REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMB_G2_PRECIPLIQRATE/
   INTEGER*4 count(536,1440,2,16,3)
   REAL*4 mean(536,1440,2,16,3)
   REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMB_G2_PRECIPTOTRATE/
   INTEGER*4 count(536,1440,2,16,3)
   REAL*4 mean(536,1440,2,16,3)
   REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE
STRUCTURE /L3CMB_G2/
  RECORD /L3CMB_G2_PRECIPTOTRATE/ precipTotRate
  RECORD /L3CMB_G2_PRECIPLIQRATE/ precipLiqRate
  RECORD /L3CMB_G2_PRECIPTOTWATERCONTENT/ precipTotWaterContent
  RECORD /L3CMB_G2_PRECIPLIQWATERCONTENT/ precipLiqWaterContent
  RECORD /L3CMB_G2_CLOUDLIQWATERCONTENT/ cloudLiqWaterContent
  RECORD /L3CMB_G2_PRECIPTOTDM/ precipTotDm
  RECORD /L3CMB_G2_PRECIPTOTLOGNW/ precipTotLogNw
  INTEGER*4 precipAllObs(536,1440,2,16)
  RECORD /L3CMB_G2_SURFPRECIPTOTRATEDIURNAL/ surfPrecipTotRateDiurnal
  INTEGER*4 surfPrecipTotRateDiurnalAllObs(536,1440,2,24)
  REAL*4 surfPrecipTotRateUn(536,1440,2)
  REAL*4 surfPrecipLiqRateUn(536,1440,2)
  REAL*4 surfPrecipTotRateProb(536,1440,2)
  REAL*4 surfPrecipLiqRateProb(536,1440,2)
END STRUCTURE

STRUCTURE /L3CMB_G1_SURFPRECIPTOTRATEDIURNAL/
  INTEGER*4 count(28,72,2,3,24)
  REAL*4 mean(28,72,2,3,24)
  REAL*4 stdev(28,72,2,3,24)
END STRUCTURE

STRUCTURE /L3CMB_G1_PRECIPTOTLOGNW/
  INTEGER*4 count(28,72,2,16,3,3)
  REAL*4 mean(28,72,2,16,3,3)
  REAL*4 stdev(28,72,2,16,3,3)
  INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMB_G1_PRECIPTOTDM/
  INTEGER*4 count(28,72,2,16,3,3)
  REAL*4 mean(28,72,2,16,3,3)
  REAL*4 stdev(28,72,2,16,3,3)
  INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMB_G1_CLOUDLIQWATERCONTENT/
  INTEGER*4 count(28,72,2,16,3)
  REAL*4 mean(28,72,2,16,3)
  REAL*4 stdev(28,72,2,16,3)
  INTEGER*4 hist(28,72,2,16,3,30)
END STRUCTURE
STRUCTURE /L3CMB_G1_PRECIPLIQWATERCONTENT/
   INTEGER*4 count(28,72,2,16,3,3)
   REAL*4 mean(28,72,2,16,3,3)
   REAL*4 stdev(28,72,2,16,3,3)
   INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMB_G1_PRECIPTOTWATERCONTENT/
   INTEGER*4 count(28,72,2,16,3,3)
   REAL*4 mean(28,72,2,16,3,3)
   REAL*4 stdev(28,72,2,16,3,3)
   INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMB_G1_PRECIPLIQRATE/
   INTEGER*4 count(28,72,2,16,3,3)
   REAL*4 mean(28,72,2,16,3,3)
   REAL*4 stdev(28,72,2,16,3,3)
   INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMB_G1_PRECIPTOTRATE/
   INTEGER*4 count(28,72,2,16,3,3)
   REAL*4 mean(28,72,2,16,3,3)
   REAL*4 stdev(28,72,2,16,3,3)
   INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMB_G1/
   RECORD /L3CMB_G1_PRECIPTOTRATE/ precipTotRate
   RECORD /L3CMB_G1_PRECIPLIQRATE/ precipLiqRate
   RECORD /L3CMB_G1_PRECIPTOTWATERCONTENT/ precipTotWaterContent
   RECORD /L3CMB_G1_PRECIPLIQWATERCONTENT/ precipLiqWaterContent
   RECORD /L3CMB_G1_CLOUDLIQWATERCONTENT/ cloudLiqWaterContent
   RECORD /L3CMB_G1_PRECIPTOTDM/ precipTotDm
   RECORD /L3CMB_G1_PRECIPTOTLOGNW/ precipTotLogNw
   INTEGER*4 precipAllObs(28,72,2,16,3)
   RECORD /L3CMB_G1_SURFPRECIPTOTRATEDIURNAL/ surfPrecipTotRateDiurnal
   INTEGER*4 surfPrecipTotRateDiurnalAllObs(28,72,2,3,24)
   REAL*4 surfPrecipTotRateUn(28,72,2)
   REAL*4 surfPrecipLiqRateUn(28,72,2)
   REAL*4 surfPrecipTotRateProb(28,72,2)
5.58 2BCMBT - Level-2 PR and TMI Combined

The Combined Level-2 product, 2BCMBT, ”Level-2 PR and TMI Combined,” is written as a one-swath structure. The swath, NS, contains 49 rays that match Ku PR. Surface variables refer to the level of the 2APR ”near surface”, not the ”estimated surface”. The following sections describe the structure and contents of the format.

Dimension definitions:

- `nscan` var: Number of scans in the granule.
- `nrayNS` 49: Number of rays (angle bins) in each NS scan.
- `nBnEnv` 10: Number of environmental bins.
- `nBnPSDlo` 9: Number of low resolution vertical range bins. The bin indexes of the low resolution PSD profile parameters are found in PSDparam-LowNode.
- `nBnPSDhi` 88: Number of high resolution vertical range bins at 250m interval.
- `nPSDlo` 2: Number of low resolution precipitation drop-size distribution parameters. Parameters are log10(Nw), mu.
- `nPSDhi` 1: Number of high resolution precipitation drop-size distribution parameters.
- `nBnTrBnd` 2: Number of bins in phase transition boundary.
- `nBnTr` 10: Number of bins in phase transition.
- `nPhsBnN` 5: Number of phase bin nodes.
- `nAB` 2: Number of power law parameters. These parameters describe particle density. The parameters are alpha and beta.
- `nemiss` 9: Number of microwave surface emissivities for TMI channels.
- `ncomp` 5: Maximum number principal components (prinComp) stored for a given observed reflectivity profile.

Figure 827 through Figure 835 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.
**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**NS** (Swath)

**NS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NS_SwathHeader</td>
<td>Metadata</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>ScanTime</td>
<td>19 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>Latitude</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>Longitude</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>scanStatus</td>
<td>24 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>navigation</td>
<td>84 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>Input</td>
<td>3038 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>aPriori</td>
<td>3136 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>surfaceAirPressure</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>surfaceAirTemperature</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>surfaceVaporDensity</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>skinTemperature</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>envParamNode</td>
<td>2 bytes</td>
<td>Array: nBnEnv x nrayNS x nscan</td>
</tr>
<tr>
<td>airPressure</td>
<td>4 bytes</td>
<td>Array: nBnEnv x nrayNS x nscan</td>
</tr>
<tr>
<td>airTemperature</td>
<td>4 bytes</td>
<td>Array: nBnEnv x nrayNS x nscan</td>
</tr>
<tr>
<td>vaporDensity</td>
<td>4 bytes</td>
<td>Array: nBnEnv x nrayNS x nscan</td>
</tr>
<tr>
<td>cloudLiqWaterCont</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>cloudIceWaterCont</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>phaseBinNodes</td>
<td>2 bytes</td>
<td>Array: nPhsBnN x nrayNS x nscan</td>
</tr>
<tr>
<td>PSDparamLowNode</td>
<td>2 bytes</td>
<td>Array: nBnPnPSDlo x nrayNS x nscan</td>
</tr>
<tr>
<td>precipTotPSDparamLow</td>
<td>4 bytes</td>
<td>Array: nPSDlo x nBnPnPSDlo x nrayNS x nscan</td>
</tr>
<tr>
<td>precipTotPSDparamHigh</td>
<td>4 bytes</td>
<td>Array: nBnPnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>precipTotWaterCont</td>
<td>4 bytes</td>
<td>Array: nBnPnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>precipTotWaterContSigma</td>
<td>4 bytes</td>
<td>Array: nBnPnPSDhi x nrayNS x nscan</td>
</tr>
</tbody>
</table>

continued on next figure

Figure 828: Data Format Structure for 2BCMBT, NS
continued from last figure

- precipTotRate 4 bytes Array: nBnPsdhi x nrayNS x nscan
- precipTotRateSigma 4 bytes Array: nBnPsdhi x nrayNS x nscan
- liqMassFracTrans 4 bytes Array: nBnTr x nrayNS x nscan
- liqRateFracTrans 4 bytes Array: nBnTr x nrayNS x nscan
- surfPrecipTotRate 4 bytes Array: nrayNS x nscan
- surfPrecipTotRateSigma 4 bytes Array: nrayNS x nscan
- surfLiqRateFrac 4 bytes Array: nrayNS x nscan
- tenMeterWindSpeed 4 bytes Array: nrayNS x nscan
- surfEmissivity 4 bytes Array: nemiss x nrayNS x nscan
- simulatedBrightTemp 4 bytes Array: nemiss x nrayNS x nscan
- nubfPIAfactor 4 bytes Array: nrayNS x nscan
- multiScatMaxContrib 4 bytes Array: nrayNS x nscan
- surfEmissSigma 4 bytes Array: nemiss x nrayNS x nscan
- tenMeterWindSigma 4 bytes Array: nrayNS x nscan
- skinTempSigma 4 bytes Array: nrayNS x nscan
- columnVaporSigma 4 bytes Array: nrayNS x nscan
- columnCloudLiqSigma 4 bytes Array: nrayNS x nscan
- errorOfDataFit 4 bytes Array: nrayNS x nscan
- pia 4 bytes Array: nrayNS x nscan
- CorrectedReflectFactor 4 bytes Array: nBnPsdhi x nrayNS x nscan
- FLG 588 bytes Group: nscan

Figure 829: Data Format Structure for 2BCMBT, NS
Figure 830: Data Format Structure for 2BCMBT, ScanTime

Figure 831: Data Format Structure for 2BCMBT, scanStatus
Figure 832: Data Format Structure for 2BCMBT, navigation
5 STANDARD GPM PRODUCTS

**Input**
- surfaceElevation 4 bytes Array: nrayNS x nscan
- surfaceType 4 bytes Array: nrayNS x nscan
- localZenithAngle 4 bytes Array: nrayNS x nscan
- precipitationFlag 4 bytes Array: nrayNS x nscan
- surfaceRangeBin 2 bytes Array: nrayNS x nscan
- lowestClutterFreeBin 2 bytes Array: nrayNS x nscan
- ellipsoidBinOffset 4 bytes Array: nrayNS x nscan
- stormTopBin 2 bytes Array: nrayNS x nscan
- stormTopAltitude 4 bytes Array: nrayNS x nscan
- zeroDegBin 2 bytes Array: nrayNS x nscan
- zeroDegAltitude 4 bytes Array: nrayNS x nscan
- precipitationType 4 bytes Array: nrayNS x nscan
- precipTypeQualityFlag 4 bytes Array: nrayNS x nscan
- piaEffective 4 bytes Array: nrayNS x nscan
- piaEffectiveSigma 4 bytes Array: nrayNS x nscan
- piaEffectiveReliabFlag 2 bytes Array: nrayNS x nscan
- sigmaZeroMeasured 4 bytes Array: nrayNS x nscan
- snowIceCover 4 bytes Array: nrayNS x nscan

**Figure 833: Data Format Structure for 2BCMBT, Input**

**aPriori**
- profClass 4 bytes Array: nrayNS x nscan
- prinComp 4 bytes Array: ncomp x nrayNS x nscan
- surfPrecipBiasRatio 4 bytes Array: nrayNS x nscan
- initNw 4 bytes Array: nBnPSDlo x nrayNS x nscan

**Figure 834: Data Format Structure for 2BCMBT, aPriori**

**FLG**
- ioQuality 4 bytes Array: nrayNS x nscan
- multiScatCalc 4 bytes Array: nrayNS x nscan
- algoType 4 bytes Array: nrayNS x nscan

**Figure 835: Data Format Structure for 2BCMBT, FLG**
Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

Millisecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: nrayNS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: nrayNS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scanStatus (Group)
**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit \( i = 1 \) and other bits = 0, the unsigned integer value is \( 2^i \)). The non-routine situations follow:
Bit Meaning if bit = 1
0   Spare (always 0)
1   SCorientation not 0 or 180
2   pointingStatus not 0
3   Non-routine limitErrorFlag
4   Non-routine operationalMode (not 1 or 11)
5   Spare (always 0)
6   Spare (always 0)
7   Spare (always 0)

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0   Latitude limit exceeded for viewed pixel locations
1   Negative scan time, invalid input
2   Error getting spacecraft attitude at scan mid-time
3   Error getting spacecraft ephemeris at scan mid-time
4   Invalid input non-unit ray vector for any pixel
5   Ray misses Earth for any pixel with normal pointing
6   Nadir calculation error for subsatellite position
7   Pixel count with geolocation error over threshold
8   Error in getting spacecraft attitude for any pixel
9   Error in getting spacecraft ephemeris for any pixel
10  Spare (always 0)
11  Spare (always 0)
12  Spare (always 0)
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits = 0 the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan): The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SConvientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan): pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
</tbody>
</table>
-8000 Non-nominal mission science orientation  
-9999 Missing  

**acsModeMidScan** (1-byte integer, array size: nscan):  
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

**targetSelectionMidScan** (1-byte integer, array size: nscan):  
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):  
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
</tbody>
</table>
limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks.
limitErrorFlag may be used in modeStatus.
Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group)

scPos (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Inertial (ECI) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector ($m s^{-1}$) of the spacecraft in ECI Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value
scLat (4-byte float, array size: nscan):
The geodetic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9  Missing value

scLon (4-byte float, array size: nscan):
The geodetic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scAttPitchGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9  Missing value
scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

Input (Group)

surfaceElevation (4-byte float, array size: nrayNS x nscan):
Altitudes above the earth ellipsoid of the surface gates from 2AKu. Values are in m. Special values are defined as:
-9999.9 Missing value
**surfaceType** (4-byte integer, array size: nrayNS x nscan):
Surface type from 2AKu. Special values are defined as:
-9999 Missing value

**localZenithAngle** (4-byte float, array size: nrayNS x nscan):
Zenith angle of the ray at the earth’s surface from 2AKu. Values are in degree. Special values are defined as:
-9999.9 Missing value

**precipitationFlag** (4-byte integer, array size: nrayNS x nscan):
Precipitation flag from 2AKu. Special values are defined as:
-9999 Missing value

**surfaceRangeBin** (2-byte integer, array size: nrayNS x nscan):
Index of the surface range bin from 2AKu. Special values are defined as:
-9999 Missing value

**lowestClutterFreeBin** (2-byte integer, array size: nrayNS x nscan):
Index of lowest clutter-free bin from 2AKu. Special values are defined as:
-9999 Missing value

**ellipsoidBinOffset** (4-byte float, array size: nrayNS x nscan):
Offset of surface bin from the earth ellipsoid from 2AKu. Values are in m. Special values are defined as:
-9999.9 Missing value

**stormTopBin** (2-byte integer, array size: nrayNS x nscan):
Index of storm top bin from 2AKu. Special values are defined as:
-9999 Missing value

**stormTopAltitude** (4-byte float, array size: nrayNS x nscan):
Altitude of storm top bin from 2AKu. Values are in m. Special values are defined as:
-9999.9 Missing value

**zeroDegBin** (2-byte integer, array size: nrayNS x nscan):
Range bin of the freezing level. Special values are defined as:
-9999 Missing value

**zeroDegAltitude** (4-byte float, array size: nrayNS x nscan):
Altitude of the freezing level. Values are in m. Special values are defined as:
-9999.9 Missing value

**precipitationType** (4-byte integer, array size: nrayNS x nscan):
Precipitation type classification from 2AKu. Special values are defined as:
-9999 Missing value

**precipTypeQualityFlag** (4-byte integer, array size: nrayNS x nscan):
Quality flag of precipitation type from 2AKu. Special values are defined as:
-9999 Missing value
**piaEffective** (4-byte float, array size: nrayNS x nscan):
Effective 2-way PIA from 2AKu. Values are in dB. Special values are defined as:
-9999.9 Missing value

**piaEffectiveSigma** (4-byte float, array size: nrayNS x nscan):
Effective PIA uncertainty from 2AKu. Values are in dB. Special values are defined as:
-9999.9 Missing value

**piaEffectiveReliabFlag** (2-byte integer, array size: nrayNS x nscan):
Reliability flag of effective PIA from 2AKu. Special values are defined as:
-9999 Missing value

**sigmaZeroMeasured** (4-byte float, array size: nrayNS x nscan):
The surface normalized radar cross section. Values range from -40 to 42 dB. Special values are defined as:
-9999.9 Missing value

**snowIceCover** (4-byte integer, array size: nrayNS x nscan):
Snow and ice cover. Values are defined as: 0 = ice-free ocean 1 = snow-free land 2 = snow-covered land 3 = sea ice. Special values are defined as:
-9999 Missing value

**aPriori** (Group)

**profClass** (4-byte integer, array size: nrayNS x nscan):
The class number of the observed reflectivity profile using a classification based upon measured reflectivity structure features. Unclassified profiles are assigned a value of -9999.

**prinComp** (4-byte float, array size: ncomp x nrayNS x nscan):
Principal components of the observed reflectivity profile, up to ncomp in number, that describe the primary modes of reflectivity structural variability. Unused principal components are assigned a value of -9999.9.

**surfPrecipBiasRatio** (4-byte float, array size: nrayNS x nscan):
The a priori ratio of mean MS-mode to NS-mode surface rain rates for the given observed reflectivity profile. Special values are defined as:
-9999.9 Missing value

**initNw** (4-byte float, array size: nBnPSDlo x nrayNS x nscan):
The initial values of the ensemble-mean, low-resolution (nBnPSDlo bins) profile of Nw associated with a given observed reflectivity profile. Nw is the intercept of the normalized gamma distribution used to describe the precipitation particle size distribution. Special values are defined as:
-9999.9 Missing value
**surfaceAirPressure** (4-byte float, array size: nrayNS x nscan):
Surface air pressure. Values range from 300 to 1100 hPa. Special values are defined as:
-9999.9 Missing value

**surfaceAirTemperature** (4-byte float, array size: nrayNS x nscan):
Surface air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

**surfaceVaporDensity** (4-byte float, array size: nrayNS x nscan):
Surface vapor density. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value

**skinTemperature** (4-byte float, array size: nrayNS x nscan):
Surface skin temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

**envParamNode** (2-byte integer, array size: nBnEnv x nrayNS x nscan):
Bin indices for environmental parameters. Special values are defined as:
-9999 Missing value

**airPressure** (4-byte float, array size: nBnEnv x nrayNS x nscan):
Air pressure. Values range from 50 to 1100 hPa. Special values are defined as:
-9999.9 Missing value

**airTemperature** (4-byte float, array size: nBnEnv x nrayNS x nscan):
Air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

**vaporDensity** (4-byte float, array size: nBnEnv x nrayNS x nscan):
Vapor density. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value

**cloudLiqWaterCont** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Cloud liquid water content. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value

**cloudIceWaterCont** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Cloud ice water content. Values range from 0 to 18 g/m³. Special values are defined as:
-9999.9 Missing value

**phaseBinNodes** (2-byte integer, array size: nPhsBnN x nrayNS x nscan):
Bin numbers indicating (0) storm top, (1) top of mixed-phase later, (2) maximum reflectivity in mixed-phase layer if bright band detected; otherwise, the freezing level from analysis, (3) bottom of mixed-phase layer, and (4) bottom of rain layer. Special values are defined as:
-9999 Missing value

**PSDparamLowNode** (2-byte integer, array size: nBnPSDlo x nrayNS x nscan):
Bin indices for low-resolution PSD parameters. Special values are defined as:
-9999 Missing value
**precipTotPSDparamLow** (4-byte float, array size: nPSDlo x nBnPSDlo x nrayNS x nscan):
Total precipitation low-resolution PSD parameters. Parameters are log10(Nw) with units log10(1 / m^4) for first value of nPSDlo, mu with no units for second value.

**precipTotPSDparamHigh** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation high-resolution PSD parameters. Values range from 0 to 20 mm_Dm. Special values are defined as:
-9999.9 Missing value

**precipTotWaterCont** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation liquid water content. Values range from 0 to 18 g/m^3. Special values are defined as:
-9999.9 Missing value

**precipTotWaterContSigma** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation liquid water content uncertainty. Values range from 0 to 18 g/m^3. Special values are defined as:
-9999.9 Missing value

**precipTotRate** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation rate. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

**precipTotRateSigma** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

**liqMassFracTrans** (4-byte float, array size: nBnTr x nrayNS x nscan):
Fraction of the precipitation mass that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

**liqRateFracTrans** (4-byte float, array size: nBnTr x nrayNS x nscan):
Fraction of the precipitation rate that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

**surfPrecipTotRate** (4-byte float, array size: nrayNS x nscan):
Surface rain rate. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value
surfPrecipTotRateSigma (4-byte float, array size: nrayNS x nscan):
Surface rain rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

surfLiqRateFrac (4-byte float, array size: nrayNS x nscan):
Surface liquid precipitation rate fraction. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

tenMeterWindSpeed (4-byte float, array size: nrayNS x nscan):
Ten meter altitude wind speed magnitude. Values range from 0 to 100 m/s. Special values are defined as:
-9999.9 Missing value

surfEmissivity (4-byte float, array size: nemiss x nrayNS x nscan):
GMI emissivities. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

simulatedBrightTemp (4-byte float, array size: nemiss x nrayNS x nscan):
GMI simulated brightness temperatures. Values range from 20 to 350 K. Special values are defined as:
-9999.9 Missing value

nubfPIAfactor (4-byte float, array size: nrayNS x nscan):
nubfPIAfactor is the factor applied to the Hitschfeld-Bordan path integrated attenuation to obtain the simulated path integrated attenuation, accounting for the nonuniform beamfilling by precipitation which is estimated from a 3x3 neighborhood of footprints. Values range from 20 to 350. Special values are defined as:
-9999.9 Missing value

multiScatMaxContrib (4-byte float, array size: nrayNS x nscan):
multiScatMaxContrib is the maximum contribution, in a given radar profile, by multiple scattering to the simulated reflectivity. Values range from 20 to 350 dB. Special values are defined as:
-9999.9 Missing value

surfEmissSigma (4-byte float, array size: nemiss x nrayNS x nscan):
Values range from 20 to 350. Special values are defined as:
-9999.9 Missing value

tenMeterWindSigma (4-byte float, array size: nrayNS x nscan):
Values range from 0 to 100 m/s. Special values are defined as:
-9999.9 Missing value

skinTempSigma (4-byte float, array size: nrayNS x nscan):
Values range from 20 to 350 K. Special values are defined as:
-9999.9 Missing value
columnVaporSigma (4-byte float, array size: nrayNS x nsan): Values range from 20 to 350 $kg/m^2$. Special values are defined as:
-9999.9 Missing value

columnCloudLiqSigma (4-byte float, array size: nrayNS x nsan): Values range from 20 to 350 $kg/m^2$. Special values are defined as:
-9999.9 Missing value

errorOfDataFit (4-byte float, array size: nrayNS x nsan): Values range from 20 to 350 K. Special values are defined as:
-9999.9 Missing value

pia (4-byte float, array size: nrayNS x nsan): Two-way path-integrated attenuation at Ku. Values range from 0 to 1000 dB. Special values are defined as:
-9999.9 Missing value

correctedReflectFactor (4-byte float, array size: nBnPSDhi x nrayNS x nsan): Corrected radar reflectivities at Ku band. Values range from -20 to 100 dBZ. Special values are defined as:
-9999.9 Missing value

FLG (Group)

ioQuality (4-byte integer, array size: nrayNS x nsan): Quality flag for input and output. The flag is a six digit number as follows.

1’s place 0 : rain estimate is valid
9 : no estimate (bad scan)

10’s place 0 : Ku data OK and rain detected using Ku
1 : Ku data OK and no rain detected using Ku
9 : bad Ku input data

100’s place 0 : Ku-SRT gives a valid PIA estimate
1 : sigma-zero at Ku is within the noise of the background
2 : sigma-zero at Ku is completely attenuated
9 : bad Ku input data

1000’s place 0 : freezing level is derived from Ku bright band
1 : freezing level is derived from GANAL analysis
9 : bad Ku input data

10000’s place 0 : Ku classified as stratiform or convective
1 : Ku classified as indeterminate
2 : precipitation not detected at Ku (no feature)
9 : bad Ku input data

100000's place  0 : some measured Tb's (interpolated to DPR grid) are valid
9 : no measured Tb's are valid

Special values are defined as:
-9999 Missing value

**multiScatCalc** (4-byte integer, array size: nrays x nsam)

Special values are defined as:
-9999 Missing value

**algoType** (4-byte integer, array size: nrays x nsam)

Special values are defined as:
-9999 Missing value

**C Structure Header file:**

```c
#ifndef _TK_2BCMBT_H_
#define _TK_2BCMBT_H_

#ifndef _L2BCMBT_FLG_
#define _L2BCMBT_FLG_

typedef struct {
    int ioQuality[49];
    int multiScatCalc[49];
    int algoType[49];
} L2BCMBT_FLG;

#endif

#ifndef _L2BCMBT_APRIORI_
#define _L2BCMBT_APRIORI_
```

typedef struct {
    int profClass[49];
    float prinComp[49][5];
    float surfPrecipBiasRatio[49];
    float initNw[49][9];
} L2BCMBT_APRIORI;

#endif

#ifndef _L2BCMBT_INPUT_
#define _L2BCMBT_INPUT_

typedef struct {
    float surfaceElevation[49];
    int surfaceType[49];
    float localZenithAngle[49];
    int precipitationFlag[49];
    short surfaceRangeBin[49];
    short lowestClutterFreeBin[49];
    float ellipsoidBinOffset[49];
    short stormTopBin[49];
    float stormTopAltitude[49];
    short zeroDegBin[49];
    float zeroDegAltitude[49];
    int precipitationType[49];
    int precipTypeQualityFlag[49];
    float piaEffective[49];
    float piaEffectiveSigma[49];
    short piaEffectiveReliabFlag[49];
    float sigmaZeroMeasured[49];
    int snowIceCover[49];
} L2BCMBT_INPUT;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
}
float scAlt;
float dprAlt;
float scAttRollGeoc;
float scAttPitchGeoc;
float scAttYawGeoc;
float scAttRollGeod;
float scAttPitchGeod;
float scAttYawGeod;
float greenHourAng;
double timeMidScan;
double timeMidScanOffset;
} NAVIGATION;

#endif

#ifndef _L2BCMBT_SCANSTATUS_
#define _L2BCMBT_SCANSTATUS_

typedef struct {
signed char dataQuality;
signed char dataWarning;
signed char missing;
signed char modeStatus;
short geoError;
short geoWarning;
short SCorientation;
short pointingStatus;
signed char acsModeMidScan;
signed char targetSelectionMidScan;
signed char operationalMode;
signed char limitErrorFlag;
double FractionalGranuleNumber;
} L2BCMBT_SCANSTATUS;

#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
short Year;
signed char Month;
signed char DayOfMonth;
} SCANTIME;

#endif
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L2BCMBT_NS_
define _L2BCMBT_NS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2BCMBT_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2BCMBT_INPUT Input;
    L2BCMBT_APRIORI aPriori;
    float surfaceAirPressure[49];
    float surfaceAirTemperature[49];
    float surfaceVaporDensity[49];
    float skinTemperature[49];
    short envParamNode[49][10];
    float airPressure[49][10];
    float airTemperature[49][10];
    float vaporDensity[49][10];
    float cloudLiqWaterCont[49][88];
    float cloudIceWaterCont[49][88];
    short phaseBinNodes[49][5];
    short PSDparamLowNode[49][9];
    float precipTotPSDparamLow[49][9][2];
    float precipTotPSDparamHigh[49][88];
    float precipTotWaterCont[49][88];
    float precipTotWaterContSigma[49][88];
    float precipTotRate[49][88];
    float precipTotRateSigma[49][88];
    float liqMassFracTrans[49][10];
    float liqRateFracTrans[49][10];
    float surfPrecipTotRate[49];
    float surfPrecipTotRateSigma[49];
float surfLiqRateFrac[49];
float tenMeterWindSpeed[49];
float surfEmissivity[49][9];
float simulatedBrightTemp[49][9];
float nubfPIAfactor[49];
float multiScatMaxContrib[49];
float surfEmissSigma[49][9];
float tenMeterWindSigma[49];
float skinTempSigma[49];
float columnVaporSigma[49];
float columnCloudLiqSigma[49];
float errorOfDataFit[49];
float pia[49];
float correctedReflectFactor[49][88];
L2BCMBT_FLG FLG;
} L2BCMBT_NS;
#endif

#ifndef _L2BCMBT_SWATHS_
define _L2BCMBT_SWATHS_

typedef struct {
   L2BCMBT_NS NS;
} L2BCMBT_SWATHS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L2BCMBT_FLG/
   INTEGER*4 ioQuality(49)
   INTEGER*4 multiScatCalc(49)
   INTEGER*4 algoType(49)
END STRUCTURE

STRUCTURE /L2BCMBT_APRIORI/
   INTEGER*4 profClass(49)
   REAL*4 prinComp(5,49)
   REAL*4 surfPrecipBiasRatio(49)
   REAL*4 initNw(9,49)
END STRUCTURE

STRUCTURE /L2BCMBT_INPUT/
  REAL*4 surfaceElevation(49)
  INTEGER*4 surfaceType(49)
  REAL*4 localZenithAngle(49)
  INTEGER*4 precipitationFlag(49)
  INTEGER*2 surfaceRangeBin(49)
  INTEGER*2 lowestClutterFreeBin(49)
  REAL*4 ellipsoidBinOffset(49)
  INTEGER*2 stormTopBin(49)
  REAL*4 stormTopAltitude(49)
  INTEGER*2 zeroDegBin(49)
  REAL*4 zeroDegAltitude(49)
  INTEGER*4 precipitationType(49)
  INTEGER*4 precipTypeQualityFlag(49)
  REAL*4 piaEffective(49)
  REAL*4 piaEffectiveSigma(49)
  INTEGER*2 piaEffectiveReliabFlag(49)
  REAL*4 sigmaZeroMeasured(49)
  INTEGER*4 snowIceCover(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
  REAL*4 scAlt
  REAL*4 dprAlt
  REAL*4 scAttRollGeoc
  REAL*4 scAttPitchGeoc
  REAL*4 scAttYawGeoc
  REAL*4 scAttRollGeod
  REAL*4 scAttPitchGeod
  REAL*4 scAttYawGeod
  REAL*4 greenHourAng
  REAL*8 timeMidScan
  REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L2BCMBT_SCANSTATUS/
  BYTE dataQuality
BYTE dataWarning
BYTE missing
BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
BYTE limitErrorFlag
REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2BCMBT_NS/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L2BCMBT_SCANSTATUS/ scanStatus
    RECORD /NAVIGATION/ navigation
    RECORD /L2BCMBT_INPUT/ Input
    RECORD /L2BCMBT_APRIORI/ aPriori
    REAL*4 surfaceAirPressure(49)
    REAL*4 surfaceAirTemperature(49)
    REAL*4 surfaceVaporDensity(49)
    REAL*4 skinTemperature(49)
    INTEGER*2 envParamNode(10,49)
    REAL*4 airPressure(10,49)
    REAL*4 airTemperature(10,49)
    REAL*4 vaporDensity(10,49)
    REAL*4 cloudLiqWaterCont(88,49)
5.59 3CMBT - Combined precipitation

3CMBT, "Combined precipitation", computes statistics of the Combined measurements at both a low horizontal resolution (G1, 5° x 5° latitude/longitude) and a high horizontal resolution (G2, 0.25° x 0.25° latitude/longitude). There will be both a monthly product and a daily product.

Units and ranges not included in this version. When units and ranges are provided and no more changes are coming then they could be added. Use specific reference for each
Dimension definitions:

- **ltL**: 28 Number of low resolution 5° grid intervals of latitude from 70°S to 70°N.
- **lnL**: 72 Number of low resolution 5° grid intervals of longitude from 180°W to 180°E.
- **ltH**: 536 Number of high resolution 0.25° grid intervals of latitude from 67°S to 67°N.
- **lnH**: 1440 Number of high resolution 0.25° grid intervals of longitude from 180°W to 180°E.
- **hgt**: 16 Number of level heights 0-15: 0: near surface, 1-10: height = 1.0km * index, 11-15: height = 10.0km + 2.0km * (index-10),
- **tim**: 24 Number of hourly local time bins.
- **rt**: 3 Number of rain types: stratiform, convective, all.
- **st**: 3 Number of surface types: ocean, land, all.
- **bin**: 30 Number of bins in histogram.

Figure 836 through Figure 854 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 837: Data Format Structure for 3CMBT, G1
Figure 838: Data Format Structure for 3CMBT, G2

Figure 839: Data Format Structure for 3CMBT, G1, precipTotRate

Figure 840: Data Format Structure for 3CMBT, G1, precipLiqRate
precipTotWaterContent
- count: 4 bytes, Array: ltL x lnL x hgt x rt x st
- mean: 4 bytes, Array: ltL x lnL x hgt x rt x st
- stdev: 4 bytes, Array: ltL x lnL x hgt x rt x st
- hist: 4 bytes, Array: ltL x lnL x hgt x rt x st x bin

Figure 841: Data Format Structure for 3CMBT, G1, precipTotWaterContent

precipLiqWaterContent
- count: 4 bytes, Array: ltL x lnL x hgt x rt x st
- mean: 4 bytes, Array: ltL x lnL x hgt x rt x st
- stdev: 4 bytes, Array: ltL x lnL x hgt x rt x st
- hist: 4 bytes, Array: ltL x lnL x hgt x rt x st x bin

Figure 842: Data Format Structure for 3CMBT, G1, precipLiqWaterContent

cloudLiqWaterContent
- count: 4 bytes, Array: ltL x lnL x hgt x st
- mean: 4 bytes, Array: ltL x lnL x hgt x st
- stdev: 4 bytes, Array: ltL x lnL x hgt x st
- hist: 4 bytes, Array: ltL x lnL x hgt x st x bin

Figure 843: Data Format Structure for 3CMBT, G1, cloudLiqWaterContent

precipTotDm
- count: 4 bytes, Array: ltL x lnL x hgt x rt x st
- mean: 4 bytes, Array: ltL x lnL x hgt x rt x st
- stdev: 4 bytes, Array: ltL x lnL x hgt x rt x st
- hist: 4 bytes, Array: ltL x lnL x hgt x rt x st x bin

Figure 844: Data Format Structure for 3CMBT, G1, precipTotDm

precipTotLogNw
- count: 4 bytes, Array: ltL x lnL x hgt x rt x st
- mean: 4 bytes, Array: ltL x lnL x hgt x rt x st
- stdev: 4 bytes, Array: ltL x lnL x hgt x rt x st
- hist: 4 bytes, Array: ltL x lnL x hgt x rt x st x bin

Figure 845: Data Format Structure for 3CMBT, G1, precipTotLogNw

surfPrecipTotRateDiurnal
- count: 4 bytes, Array: ltL x lnL x st x tim
- mean: 4 bytes, Array: ltL x lnL x st x tim
- stdev: 4 bytes, Array: ltL x lnL x st x tim

Figure 846: Data Format Structure for 3CMBT, G1, surfPrecipTotRateDiurnal
5.59 3CMBT - Combined precipitation

Figure 847: Data Format Structure for 3CMBT, G2, precipTotRate
FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

Grids (Group)
5.59 3CMBT - Combined precipitation

cloudLiqWaterContent
- count 4 bytes Array: ltH x lnH x hgt
- mean 4 bytes Array: ltH x lnH x hgt
- stdev 4 bytes Array: ltH x lnH x hgt

Figure 851: Data Format Structure for 3CMBT, G2, cloudLiqWaterContent

precipTotDm
- count 4 bytes Array: ltH x lnH x hgt x rt
- mean 4 bytes Array: ltH x lnH x hgt x rt
- stdev 4 bytes Array: ltH x lnH x hgt x rt

Figure 852: Data Format Structure for 3CMBT, G2, precipTotDm

precipTotLogNw
- count 4 bytes Array: ltH x lnH x hgt x rt
- mean 4 bytes Array: ltH x lnH x hgt x rt
- stdev 4 bytes Array: ltH x lnH x hgt x rt

Figure 853: Data Format Structure for 3CMBT, G2, precipTotLogNw

surfPrecipTotRateDiurnal
- count 4 bytes Array: ltH x lnH x tim
- mean 4 bytes Array: ltH x lnH x tim
- stdev 4 bytes Array: ltH x lnH x tim

Figure 854: Data Format Structure for 3CMBT, G2, surfPrecipTotRateDiurnal
G1 (Grid)

G1_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipTotRate (Group in G1)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitation water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipLiqRate (Group in G1)
Equivalent precipitation rate of liquid-phase precipitating water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value
**hist** (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipTotWaterContent** (Group in G1)
Equivalent water content of both liquid-phase and ice-phase precipitating water ($g/m^3$).
(Note: liquid can be in the form of rain or melt water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipLiqWaterContent** (Group in G1)
Equivalent water content of liquid-phase precipitating water ($g/m^3$). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

**count** (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value
**cloudLiqWaterContent** (Group in G1)
Equivalent water content of liquid-phase cloud water \((g/m^3)\).

**count** (4-byte integer, array size: ltL x lnL x hgt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x hgt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x hgt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x hgt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipTotDm** (Group in G1)
Volume-weighted mean of the liquid-equivalent precipitation particle diameter (mm).

**count** (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipTotLogNw** (Group in G1)
Common logarithm of the intercept of the normalized gamma distribution representing the liquid-equivalent precipitation particle size distribution \((\log_{10}(m^{-4}))\).
count (4-byte integer, array size: ltL x lnL x hgt x rt x st):
Count. Special values are defined as:
   -9999 Missing value
mean (4-byte float, array size: ltL x lnL x hgt x rt x st):
Mean. Special values are defined as:
   -9999.9 Missing value
stdev (4-byte float, array size: ltL x lnL x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
   -9999.9 Missing value
hist (4-byte integer, array size: ltL x lnL x hgt x rt x st x bin):
Histogram. Special values are defined as:
   -9999 Missing value
precipAllObs (4-byte integer, array size: ltL x lnL x hgt x st):
Number of total observations, whether precipitating or not. Special values are defined as:
   -9999 Missing value

surfPrecipTotRateDiurnal (Group in G1)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitating water in the lowest uncontaminated range-bin (mm/hr), indexed by the local time. (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x st x tim):
Count. Special values are defined as:
   -9999 Missing value
mean (4-byte float, array size: ltL x lnL x st x tim):
Mean. Special values are defined as:
   -9999.9 Missing value
stdev (4-byte float, array size: ltL x lnL x st x tim):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
   -9999.9 Missing value
surfPrecipTotRateDiurnalAllObs (4-byte integer, array size: ltL x lnL x st x tim):
Number of total diurnal observations, whether precipitating or not. Special values are defined as:
   -9999 Missing value
surfPrecipTotRateUn (4-byte float, array size: ltL x lnL):
Surface total precipitation rate unconditioned. To obtain rate conditioned on precipita-
tion, divide by the probability. Special values are defined as:
- 9999.9 Missing value

**surfPrecipLiqRateUn** (4-byte float, array size: ltL x lnL):
Surface liquid precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
- 9999.9 Missing value

**surfPrecipTotRateProb** (4-byte float, array size: ltL x lnL):
Probability of total surface precipitation. Special values are defined as:
- 9999.9 Missing value

**surfPrecipLiqRateProb** (4-byte float, array size: ltL x lnL):
Probability of liquid surface precipitation. Special values are defined as:
- 9999.9 Missing value

**G2** (Grid)

**G2_GridHeader** (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**precipTotRate** (Group in G2)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitation water (mm/hr).
(Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x hgt x rt):
Count. Special values are defined as:
- 9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x hgt x rt):
mean. Special values are defined as:
- 9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
- 9999.9 Missing value

**precipLiqRate** (Group in G2)
Equivalent precipitation rate of liquid-phase precipitating water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)
5.59  **3CMBT - Combined precipitation**

**count** (4-byte integer, array size: ltH x lnH x hgt x rt):  
Count. Special values are defined as:  
  -9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x hgt x rt):  
mean. Special values are defined as:  
  -9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x hgt x rt):  
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:  
  -9999.9  Missing value

**precipTotWaterContent** (Group in G2)  
Equivalent water content of both liquid-phase and ice-phase precipitating water ($g/m^3$). (Note: liquid can be in the form of rain or melt water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x hgt x rt):  
Count. Special values are defined as:  
  -9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x hgt x rt):  
mean. Special values are defined as:  
  -9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x hgt x rt):  
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:  
  -9999.9  Missing value

**precipLiqWaterContent** (Group in G2)  
Equivalent water content of liquid-phase precipitating water ($g/m^3$). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x hgt x rt):  
Count. Special values are defined as:  
  -9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x hgt x rt):  
mean. Special values are defined as:  
  -9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x hgt x rt):  
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

**cloudLiqWaterContent** (Group in G2)
Equivalent water content of liquid-phase cloud water \((g/m^3)\).

**count** (4-byte integer, array size: \(ltH \times lnH \times hgt\)):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: \(ltH \times lnH \times hgt\)):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: \(ltH \times lnH \times hgt\)):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipTotDm** (Group in G2)
Volume-weighted mean of the liquid-equivalent precipitation particle diameter (mm).

**count** (4-byte integer, array size: \(ltH \times lnH \times hgt \times rt\)):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: \(ltH \times lnH \times hgt \times rt\)):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: \(ltH \times lnH \times hgt \times rt\)):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipTotLogNw** (Group in G2)
Common logarithm of the intercept of the normalized gamma distribution representing the liquid-equivalent precipitation particle size distribution \((\log_{10}(m^{-4}))\).

**count** (4-byte integer, array size: \(ltH \times lnH \times hgt \times rt\)):
Count. Special values are defined as:
-9999 Missing value
**mean** (4-byte float, array size: ltH x lnH x hgt x rt):
Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipAllObs** (4-byte integer, array size: ltH x lnH x hgt):
Number of total observations, whether precipitating or not. Special values are defined as:
-9999 Missing value

---

**surfPrecipTotRateDiurnal** (Group in G2)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitating water in the lowest uncontaminated range-bin (mm/hr), indexed by the local time. (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x tim):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x tim):
Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x tim):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**surfPrecipTotRateDiurnalAllObs** (4-byte integer, array size: ltH x lnH x tim):
Number of total diurnal observations, whether precipitating or not. Special values are defined as:
-9999 Missing value

**surfPrecipTotRateUn** (4-byte float, array size: ltH x lnH):
Surface total precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9 Missing value

**surfPrecipLiqRateUn** (4-byte float, array size: ltH x lnH):
Surface liquid precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9 Missing value
surfPrecipTotRateProb (4-byte float, array size: ltH x lnH):
Probability of total surface precipitation. Special values are defined as:
-9999.9  Missing value

surfPrecipLiqRateProb (4-byte float, array size: ltH x lnH):
Probability of liquid surface precipitation. Special values are defined as:
-9999.9  Missing value

C Structure Header file:

```c
#ifndef _TK_3CMBT_H_
define _TK_3CMBT_H_

#ifndef _L3CMBT_G2_SURFPRECIPTOTRATEDIURNAL_
define _L3CMBT_G2_SURFPRECIPTOTRATEDIURNAL_

typedef struct {
    int count[24][1440][536];
    float mean[24][1440][536];
    float stdev[24][1440][536];
} L3CMBT_G2_SURFPRECIPTOTRATEDIURNAL;
#endif

#ifndef _L3CMBT_G2_PRECIPTOTLOGNW_
define _L3CMBT_G2_PRECIPTOTLOGNW_

typedef struct {
    int count[3][16][1440][536];
    float mean[3][16][1440][536];
    float stdev[3][16][1440][536];
} L3CMBT_G2_PRECIPTOTLOGNW;
#endif

#ifndef _L3CMBT_G2_PRECIPTOTDM_
define _L3CMBT_G2_PRECIPTOTDM_

typedef struct {
    int count[3][16][1440][536];
    float mean[3][16][1440][536];
    float stdev[3][16][1440][536];
} L3CMBT_G2_PRECIPTOTDM;
#endif
```
 typedef struct {
    int count[16][1440][536];
    float mean[16][1440][536];
    float stdev[16][1440][536];
} L3CMBT_G2_CLOUDLIQWATERCONTENT;

 typedef struct {
    int count[3][16][1440][536];
    float mean[3][16][1440][536];
    float stdev[3][16][1440][536];
} L3CMBT_G2_PRECIPLIQWATERCONTENT;

 typedef struct {
    int count[3][16][1440][536];
    float mean[3][16][1440][536];
    float stdev[3][16][1440][536];
} L3CMBT_G2_PRECIPTOTWATERCONTENT;

 typedef struct {
    int count[3][16][1440][536];
    float mean[3][16][1440][536];
    float stdev[3][16][1440][536];
} L3CMBT_G2_PRECIPLIQRATE;
typedef struct {
    int count[3][16][1440][536];
    float mean[3][16][1440][536];
    float stdev[3][16][1440][536];
} L3CMBT_G2_PRECIPTOTRATE;

#endif

#ifndef _L3CMBT_G2_PRECIPTOTRATE_
define _L3CMBT_G2_PRECIPTOTRATE_

typedef struct {
    L3CMBT_G2_PRECIPTOTRATE precipTotRate;
    L3CMBT_G2_PRECIPLIQRATE precipLiqRate;
    L3CMBT_G2_PRECIPTOTWATERCONTENT precipTotWaterContent;
    L3CMBT_G2_PRECIPLIQWATERCONTENT precipLiqWaterContent;
    L3CMBT_G2_CLOUDLIQWATERCONTENT cloudLiqWaterContent;
    L3CMBT_G2_PRECIPTOTDM precipTotDm;
    L3CMBT_G2_PRECIPTOTLOGNW precipTotLogNw;
    int precipAllObs[16][1440][536];
    L3CMBT_G2_SURFPRECIPTOTTRATEDIURNAL surfPrecipTotRateDiurnal;
    int surfPrecipTotRateDiurnalAllObs[24][1440][536];
    float surfPrecipTotRateUn[1440][536];
    float surfPrecipLiqRateUn[1440][536];
    float surfPrecipTotRateProb[1440][536];
    float surfPrecipLiqRateProb[1440][536];
} L3CMBT_G2;

#endif

#ifndef _L3CMBT_G1_SURFPRECIPTOTRATEDIURNAL_
define _L3CMBT_G1_SURFPRECIPTOTRATEDIURNAL_

typedef struct {
    int count[24][3][72][28];
    float mean[24][3][72][28];
    float stdev[24][3][72][28];
L3CMBT_G1_SURFPRECIPTOTRATEDIURNAL;

#endif

#ifndef _L3CMBT_G1_PRECIPTOTLOGNW_
define _L3CMBT_G1_PRECIPTOTLOGNW_

typedef struct {
    int count[3][3][16][72][28];
    float mean[3][3][16][72][28];
    float stdev[3][3][16][72][28];
    int hist[30][3][3][16][72][28];
} L3CMBT_G1_PRECIPTOTLOGNW;
#endif

#ifndef _L3CMBT_G1_PRECIPTOTDM_
define _L3CMBT_G1_PRECIPTOTDM_

typedef struct {
    int count[3][3][16][72][28];
    float mean[3][3][16][72][28];
    float stdev[3][3][16][72][28];
    int hist[30][3][3][16][72][28];
} L3CMBT_G1_PRECIPTOTDM;
#endif

#ifndef _L3CMBT_G1_CLOUDLIQWATERCONTENT_
define _L3CMBT_G1_CLOUDLIQWATERCONTENT_

typedef struct {
    int count[3][3][16][72][28];
    float mean[3][3][16][72][28];
    float stdev[3][3][16][72][28];
    int hist[30][3][3][16][72][28];
} L3CMBT_G1_CLOUDLIQWATERCONTENT;
#endif

#ifndef _L3CMBT_G1_PRECIPLIQWATERCONTENT_
define _L3CMBT_G1_PRECIPLIQWATERCONTENT_

typedef struct {
    int count[3][3][16][72][28];
    float mean[3][3][16][72][28];
    float stdev[3][3][16][72][28];
    int hist[30][3][3][16][72][28];
} L3CMBT_G1_PRECIPLIQWATERCONTENT;
#endif

5.59  3CMBT - Combined precipitation
typedef struct {
    int count[3][3][16][72][28];
    float mean[3][3][16][72][28];
    float stdev[3][3][16][72][28];
    int hist[30][3][3][16][72][28];
} L3CMBT_G1_PRECIPLIQWATERCONTENT;

#endif

#ifndef _L3CMBT_G1_PRECIPTOTWATERCONTENT_
#define _L3CMBT_G1_PRECIPTOTWATERCONTENT_

typedef struct {
    int count[3][3][16][72][28];
    float mean[3][3][16][72][28];
    float stdev[3][3][16][72][28];
    int hist[30][3][3][16][72][28];
} L3CMBT_G1_PRECIPTOTWATERCONTENT;

#endif

#ifndef _L3CMBT_G1_PRECIPLIQRATE_
#define _L3CMBT_G1_PRECIPLIQRATE_

typedef struct {
    int count[3][3][16][72][28];
    float mean[3][3][16][72][28];
    float stdev[3][3][16][72][28];
    int hist[30][3][3][16][72][28];
} L3CMBT_G1_PRECIPLIQRATE;

#endif

#ifndef _L3CMBT_G1_PRECIPTOTRATE_
#define _L3CMBT_G1_PRECIPTOTRATE_

typedef struct {
    int count[3][3][16][72][28];
    float mean[3][3][16][72][28];
    float stdev[3][3][16][72][28];
    int hist[30][3][3][16][72][28];
} L3CMBT_G1_PRECIPTOTRATE;
typedef struct {
    L3CMBT_G1_PRECIPTOTRATE precipTotRate;
    L3CMBT_G1_PRECIPLIQRATE precipLiqRate;
    L3CMBT_G1_PRECIPTOTWATERCONTENT precipTotWaterContent;
    L3CMBT_G1_PRECIPLIQWATERCONTENT precipLiqWaterContent;
    L3CMBT_G1_CLOUDLIQWATERCONTENT cloudLiqWaterContent;
    L3CMBT_G1_PRECIPTOTDM precipTotDm;
    L3CMBT_G1_PRECIPTOTLOGNW precipTotLogNw;
    int precipAllObs[3][16][72][28];
    L3CMBT_G1_SURFPRECIPTOTRATEDIURNAL surfPrecipTotRateDiurnal;
    int surfPrecipTotRateDiurnalAllObs[24][3][72][28];
    float surfPrecipTotRateUn[72][28];
    float surfPrecipLiqRateUn[72][28];
    float surfPrecipTotRateProb[72][28];
    float surfPrecipLiqRateProb[72][28];
} L3CMBT_G1;

typedef struct {
    L3CMBT_G1 G1;
    L3CMBT_G2 G2;
} L3CMBT_GRIDS;

Fortran Structure Header file:

STRUCTURE /L3CMBT_G2_SURFPRECIPTOTRATEDIURNAL/
    INTEGER*4 count(536,1440,24)
    REAL*4 mean(536,1440,24)
    REAL*4 stdev(536,1440,24)
END STRUCTURE
STRUCTURE /L3CMBT_G2_PRECIPTOTLOGNW/
   INTEGER*4 count(536,1440,16,3)
   REAL*4 mean(536,1440,16,3)
   REAL*4 stdev(536,1440,16,3)
END STRUCTURE

STRUCTURE /L3CMBT_G2_PRECIPTOTDM/
   INTEGER*4 count(536,1440,16,3)
   REAL*4 mean(536,1440,16,3)
   REAL*4 stdev(536,1440,16,3)
END STRUCTURE

STRUCTURE /L3CMBT_G2_CLOUDLIQWATERCONTENT/
   INTEGER*4 count(536,1440,16)
   REAL*4 mean(536,1440,16)
   REAL*4 stdev(536,1440,16)
END STRUCTURE

STRUCTURE /L3CMBT_G2_PRECIPLIQWATERCONTENT/
   INTEGER*4 count(536,1440,16,3)
   REAL*4 mean(536,1440,16,3)
   REAL*4 stdev(536,1440,16,3)
END STRUCTURE

STRUCTURE /L3CMBT_G2_PRECIPTOTWATERCONTENT/
   INTEGER*4 count(536,1440,16,3)
   REAL*4 mean(536,1440,16,3)
   REAL*4 stdev(536,1440,16,3)
END STRUCTURE

STRUCTURE /L3CMBT_G2_PRECIPLIQRATE/
   INTEGER*4 count(536,1440,16,3)
   REAL*4 mean(536,1440,16,3)
   REAL*4 stdev(536,1440,16,3)
END STRUCTURE

STRUCTURE /L3CMBT_G2_PRECIPTOTRATE/
   INTEGER*4 count(536,1440,16,3)
   REAL*4 mean(536,1440,16,3)
   REAL*4 stdev(536,1440,16,3)
END STRUCTURE
STRUCTURE /L3CMBT_G2/
  RECORD /L3CMBT_G2_PRECIPTOTRATE/ precipTotRate
  RECORD /L3CMBT_G2_PRECIPLIQRATE/ precipLiqRate
  RECORD /L3CMBT_G2_PRECIPTOTWATERCONTENT/ precipTotWaterContent
  RECORD /L3CMBT_G2_PRECIPLIQWATERCONTENT/ precipLiqWaterContent
  RECORD /L3CMBT_G2_CLOUDLIQWATERCONTENT/ cloudLiqWaterContent
  RECORD /L3CMBT_G2_PRECIPTOTDM/ precipTotDm
  RECORD /L3CMBT_G2_PRECIPTOTLOGNW/ precipTotLogNw
  INTEGER*4 precipAllObs(536,1440,16)
  RECORD /L3CMBT_G2_SURFPRECIPTOTRATEDIURNAL/ surfPrecipTotRateDiurnal
  INTEGER*4 surfPrecipTotRateDiurnalAllObs(536,1440,24)
  REAL*4 surfPrecipTotRateUn(536,1440)
  REAL*4 surfPrecipLiqRateUn(536,1440)
  REAL*4 surfPrecipTotRateProb(536,1440)
  REAL*4 surfPrecipLiqRateProb(536,1440)
END STRUCTURE

STRUCTURE /L3CMBT_G1_SURFPRECIPTOTRATEDIURNAL/
  INTEGER*4 count(28,72,3,24)
  REAL*4 mean(28,72,3,24)
  REAL*4 stdev(28,72,3,24)
END STRUCTURE

STRUCTURE /L3CMBT_G1_PRECIPTOTLOGNW/
  INTEGER*4 count(28,72,16,3,3)
  REAL*4 mean(28,72,16,3,3)
  REAL*4 stdev(28,72,16,3,3)
  INTEGER*4 hist(28,72,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBT_G1_PRECIPTOTDM/
  INTEGER*4 count(28,72,16,3,3)
  REAL*4 mean(28,72,16,3,3)
  REAL*4 stdev(28,72,16,3,3)
  INTEGER*4 hist(28,72,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBT_G1_CLOUDLIQWATERCONTENT/
  INTEGER*4 count(28,72,16,3)
  REAL*4 mean(28,72,16,3)
  REAL*4 stdev(28,72,16,3)
  INTEGER*4 hist(28,72,16,3,30)
END STRUCTURE
STRUCTURE /L3CMBT_G1_PRECIPLIQWATERCONTENT/
   INTEGER*4 count(28,72,16,3,3)
   REAL*4 mean(28,72,16,3,3)
   REAL*4 stdev(28,72,16,3,3)
   INTEGER*4 hist(28,72,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBT_G1_PRECIPTOTWATERCONTENT/
   INTEGER*4 count(28,72,16,3,3)
   REAL*4 mean(28,72,16,3,3)
   REAL*4 stdev(28,72,16,3,3)
   INTEGER*4 hist(28,72,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBT_G1_PRECIPLIQWATERCONTENT/
   INTEGER*4 count(28,72,16,3,3)
   REAL*4 mean(28,72,16,3,3)
   REAL*4 stdev(28,72,16,3,3)
   INTEGER*4 hist(28,72,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBT_G1_PRECIPTOTRATE/
   INTEGER*4 count(28,72,16,3,3)
   REAL*4 mean(28,72,16,3,3)
   REAL*4 stdev(28,72,16,3,3)
   INTEGER*4 hist(28,72,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBT_G1/
   RECORD /L3CMBT_G1_PRECIPTOTRATE/ precipTotRate
   RECORD /L3CMBT_G1_PRECIPLIQRATE/ precipLiqRate
   RECORD /L3CMBT_G1_PRECIPTOTWATERCONTENT/ precipTotWaterContent
   RECORD /L3CMBT_G1_PRECIPLIQWATERCONTENT/ precipLiqWaterContent
   RECORD /L3CMBT_G1_CLOUDLIQWATERCONTENT/ cloudLiqWaterContent
   RECORD /L3CMBT_G1_PRECIPTOTDM/ precipTotDm
   RECORD /L3CMBT_G1_PRECIPTOTLOGNW/ precipTotLogNw
   INTEGER*4 precipAllObs(28,72,16,3)
   RECORD /L3CMBT_G1_SURFPRECIPTOTRATEDIURNAL/ surfPrecipTotRateDiurnal
   INTEGER*4 surfPrecipTotRateDiurnalAllObs(28,72,3,24)
   REAL*4 surfPrecipTotRateUn(28,72)
   REAL*4 surfPrecipLiqRateUn(28,72)
   REAL*4 surfPrecipTotRateProb(28,72)
The Ku Level-2A product, 2AKu, "Ku precipitation," is written as a 1 swath structure. The swath is FS, full scans. The following sections describe the structure and contents of the format.

Dimension definitions:

- **nscan**: Number of scans in the granule.
- **nray**: Number of angle bins in each FS scan.
- **nbin**: Number of range bins in each NS and MS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
- **nbinSZP**: Number of range bins for sigmaZeroProfile.
- **nNP**: Number of NP kinds.
- **nearFar**: Near reference, Far reference.
- **foreBack**: Forward, Backward.
- **method**: Number of SRT methods.
- **nsdew**: Number of standard deviation effective ways.
- **nNode**: Number of binNode.
- **nDSD**: Number of DSD parameters. Parameters are dBNw and Dm (mm).
- **LS**: Liquid, solid.
- **nNUBF**: Number of NUBF parameters.
- **two**: Two.

Figure 855 through Figure 866 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 855: Data Format Structure for 2AKuX, Ku precipitation
5.60 2AKuX - Ku precipitation

![Data Format Structure for 2AKuX, ScanTime](image1)

![Data Format Structure for 2AKuX, scanStatus](image2)
Figure 858: Data Format Structure for 2AKuX, navigation
5.60 2AKuX - Ku precipitation

Figure 859: Data Format Structure for 2AKuX, PRE

Figure 860: Data Format Structure for 2AKuX, VER
Figure 861: Data Format Structure for 2AKuX, CSF
Figure 862: Data Format Structure for 2AKuX, SRT

Figure 863: Data Format Structure for 2AKuX, DSD

Figure 864: Data Format Structure for 2AKuX, Experimental
**Figure 865: Data Format Structure for 2AKuX, SLV**

**FileHeader** (Metadata): FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata): InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata): AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata): NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
5.60 2AKuX - Ku precipitation

FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

FS (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
   -99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
   -99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
   -9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
   -9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
   day. Values range from 0 to 86400 s. Special values are defined as:
   -9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
   is positive north, negative south. Values range from -90 to 90 degrees. Special values are
   defined as:
   -9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
   Longitude is positive east, negative west. A point on the 180th meridian has the value
   -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9 Missing value

scanStatus (Group)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
   meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
   i = 1 and other bits = 0, the unsigned integer value is 2**i).

   Bit Meaning if bit = 1
   0 missing
   5 geoError is not zero
   6 modeStatus is not zero
**dataWarning** (1-byte integer, array size: nsan): Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nsan): Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nsan): A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
</tbody>
</table>
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

SCoration (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCoration is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
</tbody>
</table>
2          SUNPOINT
3          GSPM (Gyro-less Sun Point)
4          MSM (Mission Science Mode)
5          SLEW
6          DELTAH
7          DELTAV
-99        UNKNOWN -- ACS mode unavailable

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
</tbody>
</table>
17    Ku/Ka Independent Standby VPRF Table OUT
18    Ku/Ka Independent Standby Phase Out
19    Ku/Ka Independent Standby Dump Out
20    Ku/Ka Independent Standby (No Science Data)

**limitErrorFlag** (1-byte integer, array size: nscan):

- Bit flags for every ray with information about echo power limit checks.
- limitErrorFlag may be used in modeStatus.
- Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule.
- Values range from 0 to 100000. Special values are defined as:
  -9999.9  Missing value

**navigation** (Group)

**scPos** (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9  Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector (m/s$^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9  Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9  Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value
scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

cAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

cAttPitchGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

cAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

cAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:

-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:

-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:

-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:

-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:

-9999.9 Missing value

**PRE** (Group)

**elevation** (4-byte float, array size: nray x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:

-9999.9 Missing value

**landSurfaceType** (4-byte integer, array size: nray x nscan):

Land surface type.

0 - 99 Ocean
100 - 199 Land
200 - 299 Coast
300 - 399 Inland water
-99999 Missing value
localZenithAngle (4-byte float, array size: nray x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values
are in degree. Special values are defined as:
-9999.9  Missing value

flagPrecip (4-byte integer, array size: nray x nscan):

Precipitation or no precipitation.

For L2 Ku and L2 Ka
0  No precipitation
1  Precipitation
-9999  Missing value

For L2 DPR
0  No precipitation by both Ku and Ka
1  Precipitation by Ka, no rain by Ku
10  Precipitation by Ku, no rain by Ka
11  Precipitation by both Ku and Ka
-9999  Missing value

flagSigmaZeroSaturation (1-byte char, array size: nray x nscan):

A flag to show whether echoPower is under a saturated
level or not at a range bin with a calculation of
sigmaZeroMeasured. Values are:

0  : normal (under saturated level)
1  : possible saturated level at real surface
2  : saturated level at real surface
99  : missing

binRealSurface (2-byte integer, array size: nray x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based
ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths,
bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the
Ellipsoid. Special values are defined as:
-9999  Missing value

binStormTop (2-byte integer, array size: nray x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based
ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:

-9999  Missing value

**heightStormTop** (4-byte float, array size: nray x nscan):
Height of storm top. Values are in m. Special values are defined as:

-9999.9  Missing value

**height** (4-byte float, array size: nbin x nray x nscan):
Height. Values are in m. Special values are defined as:

-9999.9  Missing value

**binClutterFreeBottom** (2-byte integer, array size: nray x nscan):
Range bin number for clutter free bottom. Special values are defined as:

-9999  Missing value

**sigmaZeroMeasured** (4-byte float, array size: nray x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:

-9999.9  Missing value

**zFactorMeasured** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:

-9999.9  Missing value

**ellipsoidBinOffset** (4-byte float, array size: nray x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

\[
\text{ellipsoidBinOffset} = \text{scRangeEllipsoid} - \{ \text{startBinRange} + (\text{binEllipsoid}-1) \times \text{rangeBinSize}\}
\]

scRangeEllipsoid : Distance between a sensor and the ellipsoid [m]
startBinRange : Distance between a sensor and a center of the highest observed range bin [m]
binEllipsoid : Range bin number of the Ellipsoid (1 - 260)
rangeBinSize : Range bin size [m]

-9999  Missing value

**snRatioAtRealSurface** (4-byte float, array size: nray x nscan):
Signal/Noise ratio at real surface range bin.

\[
\text{snRatioAtRealSurface} = 10 \times \log_{10}(\text{echoPowertrueV}\text{[mW]} / \text{noisePowertrueV}\text{[mW]})
\]
-9999  Missing value

**adjustFactor** (4-byte float, array size: nray x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm’) and sigmaZeroMeasured (dBs0m’).
The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
temporal adjustment for orbit number dependency.

$$dBZm' = dBZm - adjustFactor$$
$$dBs0m' = dBs0m - adjustFactor$$

**snowIceCover** (1-byte integer, array size: nray x nscan):
TBD. Special values are defined as:
-99  Missing value

**VER** (Group)

**airTemperature** (4-byte float, array size: nbin x nray x nscan):
Air Temperature. Values are in K. Special values are defined as:
-9999.9  Missing value

**binZeroDeg** (2-byte integer, array size: nray x nscan):

Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

**attenuationNP** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud
ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are
defined as:
-9999.9  Missing value

\textbf{piaNP} (4-byte float, array size: nNP x nray x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are defined as:
-9999.9  Missing value

\textbf{sigmaZeroNPCorrected} (4-byte float, array size: nray x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9  Missing value

\textbf{heightZeroDeg} (4-byte float, array size: nray x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9  Missing value

\textbf{CSF} (Group)

\textbf{flagBB} (4-byte integer, array size: nray x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.

\textbf{L2 DPR}:
0  no Bright Band
1  Bright Band detected by Ku and DFRm
2  Bright Band detected by Ku only
3  Bright Band detected by DFRm only
-1111  No rain value
-9999  Missing value

\textbf{L2 Ku and L2 Ka}:
0  BB not detected
1  BB detected
-1111  No rain value
-9999  Missing value

\textbf{binBBPeak} (2-byte integer, array size: nray x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are
defined as:
-9999 Missing value

**binBBTop** (2-byte integer, array size: nray x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binBBBottom** (2-byte integer, array size: nray x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binHeavyIcePrecipTop** (2-byte integer, array size: nray x nscan):
Range bin number for the top of heavy ice precip. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binHeavyIcePrecipBottom** (2-byte integer, array size: nray x nscan):
Range bin number for the bottom of heavy ice precip. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**nHeavyIcePrecip** (1-byte char, array size: nray x nscan):
TBD. Special values are defined as:
255 Missing value

**heightBB** (4-byte float, array size: nray x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

**widthBB** (4-byte float, array size: nray x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value
**qualityBB** (4-byte integer, array size: nray x nscan):

Quality of the bright band. When the bright band is detected, a larger positive number indicates lower confidence in the detection.

The Ku detection is clear, but the Ka and DPR detection is somewhat doubtful. The meaning of qualityBB has not been finalized.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Smeared bright band</td>
</tr>
<tr>
<td>2</td>
<td>Not so clear bright band</td>
</tr>
<tr>
<td>1</td>
<td>Clear bright band</td>
</tr>
<tr>
<td>0</td>
<td>BB not detected in the case of rain</td>
</tr>
<tr>
<td>-1111</td>
<td>No rain value</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

**typePrecip** (4-byte integer, array size: nray x nscan):

Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, convective, and other, can be obtained as follows:

When typePrecip is greater than zero,

- Major rain type = typePrecip/10000000
  - = 1  stratiform
  - = 2  convective
  - = 3  other

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1111</td>
<td>No rain value</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

Let abcdefgh be the 8 digit number,

abcdefgh

then

- a: Main rain type. (a=1,2,3),
- b: 0,
- c: 0,
- d: V rain type,
e: H rain type,
f: BB,
g: Shallow rain,
h: Small size cell.

The following numbers appear as Ku and Ka (MS/HS) rain types:

---- stratiform
1001H100
10031000

---- convective
2001H1xy (x>0 or y>0)
2002Hbxy
200310xy (x>0 or y>0)
200320xy

---- other
300330xy

where H is the rain type by H-method, and b depends on BB,
x on shallow rain and y on small size cell:

H = 1: stratiform by H-method,
  2: convective by H-method,
  3: other by H-method.

b = 0: BB not detected,
  1: BB detected.

x = 0: No shallow rain,
  1: Shallow isolated,
  3: Shallow non-isolated.

y = 0: No small size cell,
  1: Single cell,
  2: Small size cell consisting of two adjacent pixels.

In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is
also included in typePrecip and can be obtained as follows:

DFRm rain type = (typePrecip%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000 in FORTRAN

DFRm rain type
  = 1 stratiform
  = 2 convective
  = 4 transition
= 8  DFRm method cannot be applicable at Part B (in this case
the conventional method determines the major rain type)
= 9  DFRm method cannot be applicable at Part A (in this case
the conventional method determines the major rain type)

-1111  No rain value
-9999  Missing value

If dual frequency data is not available
but Ku-only or Ka-only is available,
rain type is expressed by the following 8 digit number:
  10xxxxxx --- stratiform,
  20xxxxxx --- convective,
  30xxxxxx --- other,
which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is
expressed by
  1qxxxxxx --- stratiform,
  2qxxxxxx --- convective,
  3qxxxxxx --- other,
where q>0.
Thus, by examining q, users can understand whether
data is processed by dual frequency algorithm or
single frequency algorithm.

For MS and HS, DFRm method is used.

DFRm decision classifies rain type into
  stratiform,
  convective,
and
  transition.

The DPR numbering rule can be summarized as follows:
Let opqrstuv be the 8 digit number, then

  o: Main rain type. (o=1,2,3),
  p: DFRm rain type. (p=0,1,2,4,8,9, with p=0 for single frequency data only),
  q: DFRm BB. (q=0,1),
  r: V rain type (by conventional V-method).
   Basically r=0 for inner swath and r>0 for outer swath.
   However, r>0 when only single frequency data is available,
  s: H rain type,
t: = 0 for inner swath,
    1 when BB is detected in the outer swath.
u: Shallow rain,
v: Small size cell.

DFRm type can be obtained by examining p

The meaning of p is as follows:
p = 0: single frequency data only (dual frequency data not available),
  1: stratiform by DFRm method,
  2: convective by DFRm method,
  4: transition by DFRm method,
  8: DFRm decision not available,
  9: DFRm decision not available.

Note that p>0 always in DPR processing, which is different from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, p=0 always.

The following numbers appear as DPR rain types:

* For NS outer swath *
--- stratiform
  1901H100
  19031000
--- convective
  2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
  2902Hwxy
  290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
  290320xy
--- other
  390330xy

* For NS inner swath and MS *
--- stratiform
  11B0H0xy
  14B01000
  19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
(x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
(x>0 or y>0)
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type

*************
* For HS *
*************
--- stratiform
11B0H000
14B01000
19001000 --- H decision only
--- convective
21B0H0x0 (x>0)
22B0H0x0
240010x0 (x>0, 24B010x0 with B=0)
240020x0
241010x0 (x>0, 24B010x0 with B=1)
where \( w \) depends on BB by conventional V-method, \( B \) on BB by DFRm method, \( H \) on H-method, \( x \) on shallow rain and \( y \) on small size cell:

\[
\begin{align*}
  w &= 0: \text{BB not detected by conventional V-method}, \\
      &= 1: \text{BB detected by conventional V-method}. \\
  B &= 0: \text{BB not detected by DFRm method}, \\
      &= 1: \text{BB detected by DFRm method}. \\
  H &= 1: \text{stratiform by H-method}, \\
      &= 2: \text{convective by H-method}, \\
      &= 3: \text{other by H-method}. \\
  x &= 0: \text{No shallow rain}, \\
      &= 1: \text{Shallow isolated}, \\
      &= 3: \text{Shallow non-isolated}. \\
  y &= 0: \text{No small size cell}, \\
      &= 1: \text{Single cell}, \\
      &= 2: \text{Small size cell consisting of two adjacent pixels}.
\end{align*}
\]

In the above, \( x>0 \) and \( y>0 \) are taken care of in the function \( R\_\text{type}\_\text{classification}\_\text{dpr2}() \).

---

**qualityTypePrecip** (4-byte integer, array size: nray x nscan):

Quality of the precipitation type.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Good</td>
</tr>
<tr>
<td>-1111</td>
<td>No rain value</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

**flagShallowRain** (4-byte integer, array size: nray x nscan):

Type of shallow rain

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No shallow rain</td>
</tr>
</tbody>
</table>
10  Shallow isolated (maybe)
11  Shallow isolated (certain)
20  Shallow non-isolated (maybe)
21  Shallow non-isolated (certain)
-1111 No rain value
-9999 Missing value

**flagHeavyIcePrecip** (1-byte integer, array size: nray x nscan):
This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:
- 99  Missing value

**flagAnvil** (1-byte integer, array size: nray x nscan):
flagAnvil is 1 when anvil is detected by the Ku-band radar, 0 when anvil is not detected, and -99 when the data is missing.

Note that Ka-band decision is not made because of a lower sensitivity of Ka-band radar (therefore, there does not exist any Ka-band flagAnvil; only Ku-band flagAnvil is available in Ku-only and DPR NS).

**SRT** (Group)

**pathAtten** (4-byte float, array size: nray x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
- 9999.9 Missing value

**PIAalt** (4-byte float, array size: method x nray x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where

  \[ \text{PIAalt (j=1)} = \text{PIA}_{\text{Ku}} \text{ from forward along-track spatial at kth angle bin} \]
  \[ \text{PIAalt (j=2)} = \text{PIA}_{\text{Ku}} \text{ from backward along-track spatial at kth angle bin} \]
  \[ \text{PIAalt (j=3)} = \text{PIA}_{\text{Ku}} \text{ from forward hybrid at kth angle bin} \]
  \[ \text{PIAalt (j=4)} = \text{PIA}_{\text{Ku}} \text{ from backward hybrid at kth angle bin} \]
  \[ \text{PIAalt (j=5)} = \text{PIA}_{\text{Ku}} \text{ from temporal reference at kth angle bin} \]
  \[ \text{PIAalt (j=6)} = \text{PIA}_{\text{Ku}} \text{ from light-rain temporal reference at kth angle bin} \]

Values are in dB. Special values are defined as:
- 9999.9 Missing value
PIAhb (4-byte float, array size: nray x nscan):
The 2-way attenuation of HB.

Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAhybrid (4-byte float, array size: nray x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAweight (4-byte float, array size: method x nray x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

\[ PIAweight_j = 1/\sigma_j^2 \times (1/\Sigma_j(1/\sigma_j^2)) \]

Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAweightHY (4-byte float, array size: two x nray x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

\[ PIAweight_j = 1/\sigma_j^2 \times (1/\Sigma_j(1/\sigma_j^2)) \]

Values are in dB. Special values are defined as:
-9999.9 Missing value

refScanID (2-byte integer, array size: nearFar x foreBack x nray x nscan):
The number of scan lines between the current scan and the beginning (or end) of the along-track reference data at each angle bin. The values are computed by the equation: Current Scan Number - Reference Scan Number. The values are positive for the Forward estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:

1,1 - Forward - Near reference
2,1 - Forward - Far reference
1,2 - Backward - Near reference
2,2 - Backward - Far reference
Special values are defined as:
-9999 Missing value

**reliabFactor** (4-byte float, array size: nray x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
-9999.9 Missing value

**reliabFactorAlt** (4-byte float, array size: method x nray x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAAlt. Special values are defined as:
-9999.9 Missing value

**reliabFactorHY** (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

**reliabFlag** (2-byte integer, array size: nray x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliaFactor. Reliability Flag is:

= 1 if Rel_eff > 3 ; PIAeff estimate is considered reliable
= 2 if 3 ≥ Rel_eff > 1 ; PIAeff estimate is considered marginally reliable
= 3 if Rel_eff ≤ 1 ; PIAeff is unreliable
= 4 if SNR at surface < 2dB; provides a lower bound to the path-attenuation
= 9 (no-rain case)

Special values are defined as:
-9999 Missing value

**reliabFlagHY** (2-byte integer, array size: nray x nscan):
TBD.

Special values are defined as:
-9999 Missing value

**stddevEff** (4-byte float, array size: nsdew x nray x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.
Special values are defined as:
-9999.9 Missing value

**stddevHY** (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

**zeta** (4-byte float, array size: nray x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
-9999.9 Missing value

**DSD** (Group)

**phase** (1-byte char, array size: nbin x nray x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

- phase < 100 Temperature(C)=phase-100
- phase > 200 Temperature(C)=phase-200
- phase = 100 Top of the bright band
- phase = 200 Bottom of the bright band
- phase = 125 is used for the range bins between the top and peak of bright band
- phase = 175 is used for the range bins between the peak and bottom of bright band

Integer values of phase/100 =

- 0 - solid
- 1 - mixed phase
- 2 - liquid
- 255 - Missing

**binNode** (2-byte integer, array size: nNode x nray x nscan):
The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to binRealSurface in PRE group.

For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid.
For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid.
-9999 - Missing

**Experimental (Group)**

**precipRateESurface2** (4-byte float, array size: nray x nscan):
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface2Status** (1-byte char, array size: nray x nscan):
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
255 Missing value

**sigmaZeroProfile** (4-byte float, array size: nbinszp x nray x nscan):
Surface backscattering cross section profile around the current ifov. For information on this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:
-9999.9 Missing value

**seaIceConcentration** (4-byte float, array size: nray x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact
the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value

**SLV (Group)**

**flagSLV** (1-byte integer, array size: nbin x nray x nscan):
Special values are defined as:
-99 Missing value

**paramDSD** (4-byte float, array size: nDSD x nbin x nray x nscan):
Parameters of the drop size distribution. The first index is dB\(N_w\); the second index is \(D_m\) in mm. Special values are defined as:
-9999.9 Missing value

**binEchoBottom** (2-byte integer, array size: nray x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

**piaFinal** (4-byte float, array size: nray x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

**sigmaZeroCorrected** (4-byte float, array size: nray x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9 Missing value

**zFactorCorrected** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedESurface** (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value
paramNUBF (4-byte float, array size: nNUBF x nray x nscan):
TBD. Special values are defined as:
-9999.9 Missing value

precipRate (4-byte float, array size: nbin x nray x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipWaterIntegrated (4-byte float, array size: LS x nray x nscan):
Precipitation water vertically integrated. Values are in g/m². Special values are defined as:
-9999.9 Missing value

qualitySLV (4-byte integer, array size: nray x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are
declared as:
-9999 Missing value

precipRateNearSurface (4-byte float, array size: nray x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateESurface (4-byte float, array size: nray x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are
defined as:
-9999.9 Missing value

precipRateAve24 (4-byte float, array size: nray x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values
are defined as:
-9999.9 Missing value

phaseNearSurface (1-byte char, array size: nray x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in
the DSD group at the Near-surface level. As an unsigned byte value this represents:

phaseNearSurface < 100 Temperature(C)=phaseNearSurface-100
phaseNearSurface > 200 Temperature(C)=phaseNearSurface-200
phaseNearSurface = 100 Top of the bright band
phaseNearSurface = 200 Bottom of the bright band
phaseNearSurface = 125 is used for the range bins between
the top and peak of bright band
phaseNearSurface = 175 is used for the range bins between
the peak and bottom of bright band

Integer values of phaseNearSurface/100 =
0 - solid
1 - mixed phase
2 - liquid
255 - Missing

**epsilon** (4-byte float, array size: nbin x nray x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution, 
epsilon = 1 is no adjustment. Special values are defined as:
-9999.9 Missing value

**FLG** (Group)

**flagEcho** (1-byte integer, array size: nbin x nray x nscan):
Flag of precipitation and main/side lobe clutter 
information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>

**qualityData** (4-byte integer, array size: nray x nscan):
Normal data gives ”0”. Non-zero values mean the kinds of errors. Special values are 
defined as:
-9999 Missing value

Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag 
has 2 bits of information.

The 2 bit flag for each module has values:

[higher bit lower bit]
[0 0] Good
The bits of qualityData are assigned as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 7</td>
<td>Copy of dataQuality in level 1B product</td>
</tr>
<tr>
<td>8 - 9</td>
<td>Flag by input module</td>
</tr>
<tr>
<td>10 - 11</td>
<td>Flag by preparation module</td>
</tr>
<tr>
<td>12 - 13</td>
<td>Flag by vertical module</td>
</tr>
<tr>
<td>14 - 15</td>
<td>Flag by classification module</td>
</tr>
<tr>
<td>16 - 17</td>
<td>Flag by SRT module</td>
</tr>
<tr>
<td>18 - 19</td>
<td>Flag by DSD module</td>
</tr>
<tr>
<td>20 - 21</td>
<td>Flag by solver module</td>
</tr>
<tr>
<td>22 - 23</td>
<td>Flag by output module</td>
</tr>
<tr>
<td>24 - 31</td>
<td>Spare</td>
</tr>
</tbody>
</table>

**qualityFlag** (1-byte integer, array size: nray x nscan):
Flag derived from qualityData with the following values: Special values are defined as:

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
<tr>
<td>-99</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

**flagSensor** (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>

**flagScanPattern** (2-byte integer, array size: nscan):
Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
C Structure Header file:

```c
#ifndef _TK_2AKuX_H_
define _TK_2AKuX_H_

#ifndef _L2AKuX_FLG_
define _L2AKuX_FLG_

typedef struct {
signed char flagEcho[49][176];
int qualityData[49];
signed char qualityFlag[49];
signed char flagSensor;
short flagScanPattern;
} L2AKuX_FLG;
#endif

define _L2AKuX_SLV_
define _L2AKuX_SLV_

typedef struct {
signed char flagSLV[49][176];
float paramDSD[49][176][2];
short binEchoBottom[49];
float piaFinal[49];
float sigmaZeroCorrected[49];
float zFactorCorrected[49][176];
float zFactorCorrectedESurface[49];
float zFactorCorrectedNearSurface[49];
float paramNUBF[49][3];
float precipRate[49][176];
float precipWaterIntegrated[49][2];
int qualitySLV[49];
float precipRateNearSurface[49];
float precipRateESurface[49];
float precipRateAve24[49];
unsigned char phaseNearSurface[49];
float epsilon[49][176];
} L2AKuX_SLV;
#endif
```
#ifndef _L2AKuX_EXPERIMENTAL_
#define _L2AKuX_EXPERIMENTAL_

typedef struct {
    float precipRateESurface2[49];
    unsigned char precipRateESurface2Status[49];
    float sigmaZeroProfile[49][7];
    float seaIceConcentration[49];
} L2AKuX_EXPERIMENTAL;

#endif

#ifndef _L2AKuX_DSD_
#define _L2AKuX_DSD_

typedef struct {
    unsigned char phase[49][176];
    short binNode[49][5];
} L2AKuX_DSD;

#endif

#ifndef _L2AKuX_SRT_
#define _L2AKuX_SRT_

typedef struct {
    float pathAtten[49];
    float PIAalt[49][6];
    float PIAhh[49];
    float PIAhybrid[49];
    float PIAweight[49][6];
    float PIAweightHY[49][2];
    short refScanID[49][2][2];
    float reliabFactor[49];
    float reliabFactorAlt[49][6];
    float reliabFactorHY[49];
    short reliabFlag[49];
    short reliabFlagHY[49];
    float stddevEff[49][3];
    float stddevHY[49];
    float zeta[49];
} L2AKuX_SRT;
#ifndef _L2AKuX_CSF_
#define _L2AKuX_CSF_

typedef struct {
    int flagBB[49];
    short binBBPeak[49];
    short binBBTop[49];
    short binBBBottom[49];
    short binHeavyIcePrecipTop[49];
    short binHeavyIcePrecipBottom[49];
    unsigned char nHeavyIcePrecip[49];
    float heightBB[49];
    float widthBB[49];
    int qualityBB[49];
    int typePrecip[49];
    int qualityTypePrecip[49];
    int flagShallowRain[49];
    signed char flagHeavyIcePrecip[49];
    signed char flagAnvil[49];
} L2AKuX_CSF;
#endif

#ifndef _L2AKuX_VER_
#define _L2AKuX_VER_

typedef struct {
    float airTemperature[49][176];
    short binZeroDeg[49];
    float attenuationNP[49][176];
    float piaNP[49][4];
    float sigmaZeroNPCorrected[49];
    float heightZeroDeg[49];
} L2AKuX_VER;
#endif

#ifndef _L2AKuX_PRE_
#define _L2AKuX_PRE_

typedef struct {

float elevation[49];
int landSurfaceType[49];
float localZenithAngle[49];
int flagPrecip[49];
unsigned char flagSigmaZeroSaturation[49];
short binRealSurface[49];
short binStormTop[49];
float heightStormTop[49];
float height[49][176];
short binClutterFreeBottom[49];
float sigmaZeroMeasured[49];
float zFactorMeasured[49][176];
float ellipsoidBinOffset[49];
float snRatioAtRealSurface[49];
float adjustFactor[49];
signed char snowIceCover[49];
} L2AKuX_PRE;

#endif

#elif _NAVIGATION_
#define _NAVIGATION_

typedef struct {
  float scPos[3];
  float scVel[3];
  float scLat;
  float scLon;
  float scAlt;
  float dprAlt;
  float scAttRollGeoc;
  float scAttPitchGeoc;
  float scAttYawGeoc;
  float scAttRollGeod;
  float scAttPitchGeod;
  float scAttYawGeod;
  float greenHourAng;
  double timeMidScan;
  double timeMidScanOffset;
} NAVIGATION;

#endif
#ifndef _L2AKuX_SCANSTATUS_
define _L2AKuX_SCANSTATUS_

typedef struct {
signed char dataQuality;
signed char dataWarning;
signed char missing;
signed char modeStatus;
short geoError;
short geoWarning;
short SCorientation;
short pointingStatus;
signed char acsModeMidScan;
signed char targetSelectionMidScan;
signed char operationalMode;
signed char limitErrorFlag;
double FractionalGranuleNumber;
} L2AKuX_SCANSTATUS;
#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
short Year;
signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L2AKuX_FS_
define _L2AKuX_FS_

typedef struct {
   SCANTIME ScanTime;

#endif
#endif
#endif
#endif
float Latitude[49];
float Longitude[49];
L2AKuX_SCANSTATUS scanStatus;
NAVIGATION navigation;
L2AKuX_PRE PRE;
L2AKuX_VER VER;
L2AKuX_CSR CSF;
L2AKuX_SRT SRT;
L2AKuX_DSD DSD;
L2AKuX_EXPERIMENTAL Experimental;
L2AKuX_SLV SLV;
L2AKuX_FLG FLG;
}
L2AKuX_FS;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L2AKuX_FLG/
  BYTE flagEcho(176,49)
  INTEGER*4 qualityData(49)
  BYTE qualityFlag(49)
  BYTE flagSensor
  INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2AKuX_SLV/
  BYTE flagSLV(176,49)
  REAL*4 paramDSD(2,176,49)
  INTEGER*2 binEchoBottom(49)
  REAL*4 piaFinal(49)
  REAL*4 sigmaZeroCorrected(49)
  REAL*4 zFactorCorrected(176,49)
  REAL*4 zFactorCorrectedESurface(49)
  REAL*4 zFactorCorrectedNearSurface(49)
  REAL*4 paramNUBF(3,49)
  REAL*4 precipRate(176,49)
  REAL*4 precipWaterIntegrated(2,49)
  INTEGER*4 qualitySLV(49)
  REAL*4 precipRateNearSurface(49)
  REAL*4 precipRateESurface(49)
REAL*4 precipRateAve24(49)
CHARACTER phaseNearSurface(49)
REAL*4 epsilon(176,49)
END STRUCTURE

STRUCTURE /L2AKuX_EXPERIMENTAL/
    REAL*4 precipRateESurface2(49)
    CHARACTER precipRateESurface2Status(49)
    REAL*4 sigmaZeroProfile(7,49)
    REAL*4 seaIceConcentration(49)
END STRUCTURE

STRUCTURE /L2AKuX_DSD/
    CHARACTER phase(176,49)
    INTEGER*2 binNode(5,49)
END STRUCTURE

STRUCTURE /L2AKuX_SRT/
    REAL*4 pathAtten(49)
    REAL*4 PIAalt(6,49)
    REAL*4 PIAhb(49)
    REAL*4 PIAhybrid(49)
    REAL*4 PIAweight(6,49)
    REAL*4 PIAweightHY(2,49)
    INTEGER*2 refScanID(2,2,49)
    REAL*4 reliabFactor(49)
    REAL*4 reliabFactorAlt(6,49)
    REAL*4 reliabFactorHY(49)
    INTEGER*2 reliabFlag(49)
    INTEGER*2 reliabFlagHY(49)
    REAL*4 stddevEff(3,49)
    REAL*4 stddevHY(49)
    REAL*4 zeta(49)
END STRUCTURE

STRUCTURE /L2AKuX_CSF/
    INTEGER*4 flagBB(49)
    INTEGER*2 binBBPeak(49)
    INTEGER*2 binBBTop(49)
    INTEGER*2 binBBBottom(49)
    INTEGER*2 binHeavyIcePrecipTop(49)
    INTEGER*2 binHeavyIcePrecipBottom(49)
    CHARACTER nHeavyIcePrecip(49)
REAL*4 heightBB(49)
REAL*4 widthBB(49)
INTEGER*4 qualityBB(49)
INTEGER*4 typePrecip(49)
INTEGER*4 qualityTypePrecip(49)
INTEGER*4 flagShallowRain(49)
BYTE flagHeavyIcePrecip(49)
BYTE flagAnvil(49)
END STRUCTURE

STRUCTURE /L2AKuX_VER/
  REAL*4 airTemperature(176,49)
  INTEGER*2 binZeroDeg(49)
  REAL*4 attenuationNP(176,49)
  REAL*4 piaNP(4,49)
  REAL*4 sigmaZeroNPCorrected(49)
  REAL*4 heightZeroDeg(49)
END STRUCTURE

STRUCTURE /L2AKuX_PRE/
  REAL*4 elevation(49)
  INTEGER*4 landSurfaceType(49)
  REAL*4 localZenithAngle(49)
  INTEGER*4 flagPrecip(49)
  CHARACTER flagSigmaZeroSaturation(49)
  INTEGER*2 binRealSurface(49)
  INTEGER*2 binStormTop(49)
  REAL*4 heightStormTop(49)
  REAL*4 height(176,49)
  INTEGER*2 binClutterFreeBottom(49)
  REAL*4 sigmaClutterFreeBottom(49)
  REAL*4 zFactorMeasured(49)
  REAL*4 ellipsoidBinOffset(49)
  REAL*4 snRatioAtRealSurface(49)
  REAL*4 adjustFactor(49)
  BYTE snowIceCover(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
REAL*4 scAlt
REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L2AKuX_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AKuX_FS/
  RECORD /SCANTIME/ ScanTime
5.61 2AKaX - Ka precipitation

The Ka Level-2A product, 2AKaX, "Ka precipitation," is written as a 2 swath structure. The swaths are FS, full scans, and HS, high sensitivity scans. The following sections describe the structure and contents of the format.

Dimension definitions:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>nscan</td>
<td>Number of scans in the granule.</td>
<td></td>
</tr>
<tr>
<td>nray</td>
<td>Number of angle bins in each FS scan.</td>
<td>49</td>
</tr>
<tr>
<td>nrayHS</td>
<td>Number of angle bins in each HS scan.</td>
<td>24</td>
</tr>
<tr>
<td>nbin</td>
<td>Number of range bins in each FS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.</td>
<td>176</td>
</tr>
<tr>
<td>nbinHS</td>
<td>Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.</td>
<td>88</td>
</tr>
<tr>
<td>nbinSZP</td>
<td>Number of range bins for sigmaZeroProfile.</td>
<td>7</td>
</tr>
<tr>
<td>nbinSZPHS</td>
<td>Number of range bins for sigmaZeroProfile in each HS scan.</td>
<td>5</td>
</tr>
<tr>
<td>nNP</td>
<td>Number of NP kinds.</td>
<td>4</td>
</tr>
<tr>
<td>nearFar</td>
<td>Near reference, Far reference.</td>
<td>2</td>
</tr>
<tr>
<td>foreBack</td>
<td>Forward, Backward.</td>
<td>2</td>
</tr>
<tr>
<td>method</td>
<td>Number of SRT methods.</td>
<td>6</td>
</tr>
<tr>
<td>nsdew</td>
<td>Number of standard deviation effective ways.</td>
<td>3</td>
</tr>
<tr>
<td>nNode</td>
<td>Number of binNode.</td>
<td>5</td>
</tr>
<tr>
<td>nDSD</td>
<td>Number of DSD parameters. Parameters are dBNw and Dm (mm).</td>
<td>2</td>
</tr>
<tr>
<td>LS</td>
<td>Liquid, solid.</td>
<td>2</td>
</tr>
<tr>
<td>nNUBF</td>
<td>Number of NUBF parameters.</td>
<td>3</td>
</tr>
<tr>
<td>two</td>
<td>Two.</td>
<td>2</td>
</tr>
<tr>
<td>three</td>
<td>Number 3.</td>
<td>3</td>
</tr>
</tbody>
</table>

Figure 867 through Figure 891 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the
Figure 867: Data Format Structure for 2AKaX, Ka precipitation

Fortran Structure Header File.
5.61 2AKaX - Ka precipitation

Figure 868: Data Format Structure for 2AKaX, FS

Figure 869: Data Format Structure for 2AKaX, HS
Figure 870: Data Format Structure for 2AKaX, FS, ScanTime

Figure 871: Data Format Structure for 2AKaX, FS, scanStatus
Figure 872: Data Format Structure for 2AKaX, FS, navigation
Figure 873: Data Format Structure for 2AKaX, FS, PRE

Figure 874: Data Format Structure for 2AKaX, FS, VER
Figure 875: Data Format Structure for 2AKaX, FS, CSF
Figure 876: Data Format Structure for 2AKaX, FS, SRT

Figure 877: Data Format Structure for 2AKaX, FS, DSD

Figure 878: Data Format Structure for 2AKaX, FS, Experimental
Figure 879: Data Format Structure for 2AKaX, FS, SLV

Figure 880: Data Format Structure for 2AKaX, FS, FLG
Figure 881: Data Format Structure for 2AKaX, HS, ScanTime

Figure 882: Data Format Structure for 2AKaX, HS, scanStatus
Figure 883: Data Format Structure for 2AKaX, HS, navigation
Figure 884: Data Format Structure for 2AKaX, HS, PRE

Figure 885: Data Format Structure for 2AKaX, HS, VER
Figure 886: Data Format Structure for 2AKaX, HS, CSF
Figure 887: Data Format Structure for 2AKaX, HS, SRT

Figure 888: Data Format Structure for 2AKaX, HS, DSD
5.61 2AKaX - Ka precipitation

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

AlgorithmRuntimeInfo (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

FS (Swath)

FS_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in FS)
A UTC time associated with the scan.
Figure 890: Data Format Structure for 2AKaX, HS, SLV

Figure 891: Data Format Structure for 2AKaX, HS, FLG
Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
scanStatus (Group in FS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$, the unsigned integer value is $2^i$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Nominal pointing in Mission Science Mode
GPS point solution stale and PVT ephemeris used
GEONS solution stale and GEONS ephemeris used
Non-nominal mission science orientation
Missing

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

Value Meaning
0 LAUNCH
1 RATENULL
2 SUNPOINT
3 GSPM (Gyro-less Sun Point)
4 MSM (Mission Science Mode)
5 SLEW
6 DELTAH
7 DELTAV
-99 UNKNOWN -- ACS mode unavailable

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

Value Meaning
0 S/C Z axis nadir, +X in flight direction
1 Flight Z axis nadir, +X in flight direction
2 S/C Z axis nadir, -X in flight direction
3 Flight Z axis nadir, -X in flight direction
4 +90 yaw for DPR antenna pattern calibration
5 -90 yaw for DPR antenna pattern calibration
-99 Missing

operationalMode (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry.
operationalMode is used in modeStatus. The range is 1 to 20.

Value Meaning
1 Ku/Ka Observation
2 Ku/Ka External Calibration
3 Ku/Ka Internal Calibration
limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. LimitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group in FS)

scPos (4-byte float, array size: XYZ x nscan):
The position vector(m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value
**5.61  2AKaX - Ka precipitation**

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector \((m s^{-1})\) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from \(-10000000\) to \(10000000\) m/s. Special values are defined as:
-9999.9  Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9  Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value
scAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

timeMidScan (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

timeMidScanOffset (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

PRE (Group in FS)
**elevation** (4-byte float, array size: nray x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product.
Values are in m. Special values are defined as:
-9999.9Missing value

**landSurfaceType** (4-byte integer, array size: nray x nscan):

Land surface type.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 99</td>
<td>Ocean</td>
</tr>
<tr>
<td>100 - 199</td>
<td>Land</td>
</tr>
<tr>
<td>200 - 299</td>
<td>Coast</td>
</tr>
<tr>
<td>300 - 399</td>
<td>Inland water</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

**localZenithAngle** (4-byte float, array size: nray x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values are in degree. Special values are defined as:
-9999.9Missing value

**flagPrecip** (4-byte integer, array size: nray x nscan):
Precipitation or no precipitation.

For L2 Ku and L2 Ka

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No precipitation</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

For L2 DPR

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No precipitation by both Ku and Ka</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation by Ka, no rain by Ku</td>
</tr>
<tr>
<td>10</td>
<td>Precipitation by Ku, no rain by Ka</td>
</tr>
<tr>
<td>11</td>
<td>Precipitation by both Ku and Ka</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing value</td>
</tr>
</tbody>
</table>

**flagSigmaZeroSaturation** (1-byte char, array size: nray x nscan):
A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of
sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nray x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nray x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

heightStormTop (4-byte float, array size: nray x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value

height (4-byte float, array size: nbin x nray x nscan):
Height. Values are in m. Special values are defined as:
-9999.9 Missing value

binClutterFreeBottom (2-byte integer, array size: nray x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

sigmaZeroMeasured (4-byte float, array size: nray x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:
-9999.9 Missing value

zFactorMeasured (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:
-9999.9 Missing value

ellipsoidBinOffset (4-byte float, array size: nray x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

ellipsoidBinOffset =
   scRangeEllipsoid - { startBinRange + (binEllipsoid-1) x rangeBinSize}
scRangeEllipsoid : Distance between a sensor and the ellipsoid [m]
startBinRange : Distance between a sensor and a center
    of the highest observed range bin [m]
binEllipsoid : Range bin number of the Ellipsoid (1 - 260)
rangleBinSize : Range bin size [m]
-9999  Missing value

**snRatioAtRealSurface** (4-byte float, array size: nray x nscan):
Signal/Noise ratio at real surface range bin.

\[
\text{snRatioAtRealSurface} = 10. \times \log_{10}(\text{echoPowertrueV[mW]}/\text{noisePowertrueV[mW]})
\]
-9999  Missing value

**adjustFactor** (4-byte float, array size: nray x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm’) and sigmaZeroMeasured (dBs0m’).

\[\text{dBZm’} = \text{dBZm} - \text{adjustFactor}\]
\[\text{dBs0m’} = \text{dBs0m} - \text{adjustFactor}\]
The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
temporal adjustment for orbit number dependency.

**snowIceCover** (1-byte integer, array size: nray x nscan):
TBD. Special values are defined as:
-99  Missing value

**VER** (Group in FS)

**airTemperature** (4-byte float, array size: nbin x nray x nscan):
Air Temperature. Values are in K. Special values are defined as:
-9999.9  Missing value

**binZeroDeg** (2-byte integer, array size: nray x nscan):
Range bin number with 0 degrees C level.
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid.
For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

*attenuationNP* (4-byte float, array size: nbin x nray x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are defined as:
-9999.9 Missing value

*piaNP* (4-byte float, array size: nNP x nray x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are defined as:
-9999.9 Missing value

*sigmaZeroNPCorrected* (4-byte float, array size: nray x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

*heightZeroDeg* (4-byte float, array size: nray x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9 Missing value

CSF (Group in FS)

*flagBB* (4-byte integer, array size: nray x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.

L2 DPR:
0 no Bright Band
1 Bright Band detected by Ku and DFRm
2 Bright Band detected by Ku only
3 Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value

L2 Ku and L2 Ka:
0 BB not detected
1 BB detected
-1111 No rain value
-9999 Missing value

**binBBPeak** (2-byte integer, array size: nray x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binBBTop** (2-byte integer, array size: nray x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binBBBottom** (2-byte integer, array size: nray x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binHeavyIcePrecipTop** (2-byte integer, array size: nray x nscan):
Range bin number for the top of heavy ice precip. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binHeavyIcePrecipBottom** (2-byte integer, array size: nray x nscan):
Range bin number for the bottom of heavy ice precip. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with
88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

\texttt{nHeavyIcePrecip} (1-byte char, array size: nray x nscan):
TBD. Special values are defined as:
255 Missing value

\texttt{heightBB} (4-byte float, array size: nray x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

\texttt{widthBB} (4-byte float, array size: nray x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

\texttt{qualityBB} (4-byte integer, array size: nray x nscan):
Quality of the bright band.
When the bright band is detected,
a larger positive number indicates lower confidence in the detection.

The Ku detection is clear, but the Ka and DPR detection is somewhat doubtful.

The meaning of qualityBB has not been finalized.

3 Smeread bright band
2 Not so clear bright band
1 Clear bright band
0 BB not detected in the case of rain
-1111 No rain value
-9999 Missing value

\texttt{typePrecip} (4-byte integer, array size: nray x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, onvective, and other, can be obtained as follows:

When typePrecip is greater than zero,
Major rain type = typePrecip/10000000
= 1 stratiform
= 2  convective
= 3  other

-1111  No rain value
-9999  Missing value

Let abcdefgh be the 8 digit number,

abcdefgh

then

a: Main rain type. (a=1,2,3),
b: 0,
c: 0,
d: V rain type,
e: H rain type,
f: BB,
g: Shallow rain,
h: Small size cell.

The following numbers appear as Ku and Ka (MS/HS) rain types:

---- stratiform
1001H100
10031000

---- convective
2001H1xy (x>0 or y>0)
2002Hbxy
200310xy (x>0 or y>0)
200320xy

---- other
300330xy

where H is the rain type by H-method, and b depends on BB,
x on shallow rain and y on small size cell:

H = 1: stratiform by H-method,
    2: convective by H-method,
    3: other by H-method.

b = 0: BB not detected,
    1: BB detected.

x = 0: No shallow rain,
    1: Shallow isolated,
    3: Shallow non-isolated.
y = 0: No small size cell,
    1: Single cell,
    2: Small size cell consisting of two adjacent pixels.
============================================================= 
In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:

DFRm rain type = (typePrecip%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000 in FORTRAN

DFRm rain type
    = 1    stratiform
    = 2    convective
    = 4    transition
    = 8    DFRm method cannot be applicable at Part B (in this case the conventional method determines the major rain type)
    = 9    DFRm method cannot be applicable at Part A (in this case the conventional method determines the major rain type)
-1111 No rain value
-9999 Missing value

If dual frequency data is not available but Ku-only or Ka-only is available, rain type is expressed by the following 8 digit number:
    10xxxxxx --- stratiform,
    20xxxxxx --- convective,
    30xxxxxx --- other,
which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by
    1qxxxxxx --- stratiform,
    2qxxxxxx --- convective,
    3qxxxxxx --- other,
where q>0.
Thus, by examining q, users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.
============================================================= 
For MS and HS, DFRm method is used.
============================================================= 
DFRm decision classifies rain type into
    stratiform,
    convective,
    and
    transition.

The DPR numbering rule can be summarized as follows:
Let opqrstuv be the 8 digit number, then
  o: Main rain type. (o=1,2,3),
  p: DFRm rain type. (p=0,1,2,4,8,9, with p=0 for single frequency data only),
  q: DFRm BB. (q=0,1),
  r: V rain type (by conventional V-method).
    Basically r=0 for inner swath and r>0 for outer swath.
    However, r>0 when only single frequency data is available,
  s: H rain type,
  t: = 0 for innner swath,
      1 when BB is detected in the outer swath.
  u: Shallow rain,
  v: Small size cell.

DFRm type can be obtained by examining p

The meaning of p is as follows:
p = 0: single frequency data only (dual frequncy data not available),
  1: stratiform by DFRm method,
  2: convective by DFRm method,
  4: transition by DFRm method,
  8: DFRm decision not available,
  9: DFRm decision not available.

Note that p>0 always in DPR processing, which is different
from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, p=0 always.

The following numbers appear as DPR rain types:
--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy
--- other
390330xy

*******************************
* For NS inner swath and MS *
*******************************

--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type.
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
(x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
(x>0 or y>0)
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type

**************
* For HS  *
**************

--- stratiform
11B0H000
14B01000
19001000 --- H decision only

--- convective
21B0H0x0 (x>0)
22B0H0x0
240010x0 (x>0, 24B010x0 with B=0)
240020x0
241010x0 (x>0, 24B010x0 with B=1)
290010x0 (x>0) --- H decision only
290020x0 --- H decision only

--- other
340030x0
390030x0 --- H decision only

where w depends on BB by conventional V-method, B on BB by DFRm method, H on H-method, x on shallow rain and y on small size cell:

w = 0: BB not detected by conventional V-method,
1: BB detected by conventional V-method.

B = 0: BB not detected by DFRm method,
1: BB detected by DFRm method.

H = 1: stratiform by H-method,
2: convective by H-method,
3: other by H-method.

x = 0: No shallow rain,
1: Shallow isolated,
3: Shallow non-isolated.

y = 0: No small size cell,
1: Single cell,
2: Small size cell consisting of two adjacent pixels.

In the above, x>0 and y>0 are taken care of in the function R\_type\_classification\_dpr2().
qualityTypePrecip (4-byte integer, array size: nray x nsscan):

Quality of the precipitation type.

1       Good
-1111   No rain value
-9999   Missing value

flagShallowRain (4-byte integer, array size: nray x nsscan):

Type of shallow rain
0       No shallow rain
10      Shallow isolated (maybe)
11      Shallow isolated (certain)
20      Shallow non-isolated (maybe)
21      Shallow non-isolated (certain)
-1111   No rain value
-9999   Missing value

flagHeavyIcePrecip (1-byte integer, array size: nray x nsscan):
This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:
-99       Missing value

SRT (Group in FS)

pathAtten (4-byte float, array size: nray x nsscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
-9999.9   Missing value

PIAalt (4-byte float, array size: method x nray x nsscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where

PIAalt (j=1) = PIA_Ku from forward along-track spatial at kth angle bin
PIAalt (j=2) = PIA_Ku from backward along-track spatial at kth angle bin
PIAalt (j=3) = PIA_Ku from forward hybrid at kth angle bin
PIAalt (j=4) = PIA_Ku from backward hybrid at kth angle bin
PIAalt (j=5) = PIA_Ku from temporal reference at kth angle bin
PIAalt (j=6) = PIA_Ku from light-rain temporal reference at kth angle bin
Values are in dB. Special values are defined as:
-9999.9  Missing value

**PIAhb** (4-byte float, array size: nray x nscan):  
The 2-way attenuation of HB.

Values are in dB. Special values are defined as:
-9999.9  Missing value

**PIAhybrid** (4-byte float, array size: nray x nscan):  
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as:
-9999.9  Missing value

**PIAweight** (4-byte float, array size: method x nray x nscan):  
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where \( j \) is method and \( \sigma_j \) is the standard deviation of reference data for method \( j \).

\[
PIAweight_j = \frac{1}{\sigma_j^2} \times \left( \frac{1}{\sum_j (1/\sigma_j^2)} \right)
\]

Values are in dB. Special values are defined as:
-9999.9  Missing value

**PIAweightHY** (4-byte float, array size: two x nray x nscan):  
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where \( j \) is method and \( \sigma_j \) is the standard deviation of reference data for method \( j \).

\[
PIAweight_{jHY} = \frac{1}{\sigma_j^2} \times \left( \frac{1}{\sum_j (1/\sigma_j^2)} \right)
\]

Values are in dB. Special values are defined as:
-9999.9  Missing value

**refScanID** (2-byte integer, array size: nearFar x foreBack x nray x nscan):  
The number of scan lines between the current scan and the beginning (or end) of the along-track reference data at each angle bin. The values are computed by the equation: Current Scan Number - Reference Scan Number. The values are positive for the Forward estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:

1,1 - Forward - Near reference  
2,1 - Forward - Far reference  
1,2 - Backward - Near reference  
2,2 - Backward - Far reference
Special values are defined as:
-9999 Missing value

**reliabFactor** (4-byte float, array size: nray x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
-9999.9 Missing value

**reliabFactorAlt** (4-byte float, array size: method x nray x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAAlt. Special values are defined as:
-9999.9 Missing value

**reliabFactorHY** (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

**reliabFlag** (2-byte integer, array size: nray x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:
- 1 if Rel_eff > 3; PIAeff estimate is considered reliable
- 2 if 3 ≥ Rel_eff > 1; PIAeff estimate is considered marginally reliable
- 3 if Rel_eff ≤ 1; PIAeff is unreliable
- 4 if SNR at surface < 2dB; provides a lower bound to the path-attenuation
- 9 (no-rain case)

Special values are defined as:
-9999 Missing value

**reliabFlagHY** (2-byte integer, array size: nray x nscan):
TBD.

Special values are defined as:
-9999 Missing value

**stddevEff** (4-byte float, array size: nsdew x nray x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.
Special values are defined as:
-9999.9  Missing value

**stddevHY** (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9  Missing value

**zeta** (4-byte float, array size: nray x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
-9999.9  Missing value

**DSD** (Group in FS)

**phase** (1-byte char, array size: nbin x nray x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

- phase < 100  Temperature(C)=phase-100
- phase > 200  Temperature(C)=phase-200
- phase = 100  Top of the bright band
- phase = 200  Bottom of the bright band
- phase = 125 is used for the range bins between
  the top and peak of bright band
- phase = 175 is used for the range bins between
  the peak and bottom of bright band

Integer values of phase/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

**binNode** (2-byte integer, array size: nNode x nray x nscan):
The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to binRealSurface in PRE group.

For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
-9999 - Missing

**Experimental (Group in FS)**

*precipRateESurface2* (4-byte float, array size: nray x nscan):
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

*precipRateESurface2Status* (1-byte char, array size: nray x nscan):
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
255 Missing value

*sigmaZeroProfile* (4-byte float, array size: nbinSZP x nray x nscan):
Surface backscattering cross section profile around the current ifov. For information on this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:
-9999.9 Missing value

*seaIceConcentration* (4-byte float, array size: nray x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact
the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value

**SLV** (Group in FS)

**flagSLV** (1-byte integer, array size: nbin x nray x nscan):
Special values are defined as:
-99 Missing value

**paramDSD** (4-byte float, array size: nDSD x nbin x nray x nscan):
Parameters of the drop size distribution. The first index is dBNw; the second index is
Dm in mm. Special values are defined as:
-9999.9 Missing value

**binEchoBottom** (2-byte integer, array size: nray x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data
window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from
1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

**piaFinal** (4-byte float, array size: nray x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Val-
ues are in dB. Special values are defined as:
-9999.9 Missing value

**sigmaZeroCorrected** (4-byte float, array size: nray x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special
values are defined as:
-9999.9 Missing value

**zFactorCorrected** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special
values are defined as:
-9999.9 Missing value

**zFactorCorrectedESurface** (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special
values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nray x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special
values are defined as:
-9999.9 Missing value
paramNUBF (4-byte float, array size: nNUBF x nray x nscan):
TBD. Special values are defined as:
-9999.9 Missing value

precipRate (4-byte float, array size: nbin x nray x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipWaterIntegrated (4-byte float, array size: LS x nray x nscan):
Precipitation water vertically integrated. Values are in g/m². Special values are defined as:
-9999.9 Missing value

qualitySLV (4-byte integer, array size: nray x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are defined as:
-9999 Missing value

precipRateNearSurface (4-byte float, array size: nray x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateESurface (4-byte float, array size: nray x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateAve24 (4-byte float, array size: nray x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

phaseNearSurface (1-byte char, array size: nray x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

phaseNearSurface < 100 Temperature(C)=phaseNearSurface-100
phaseNearSurface > 200 Temperature(C)=phaseNearSurface-200
phaseNearSurface = 100 Top of the bright band
phaseNearSurface = 200 Bottom of the bright band
phaseNearSurface = 125 is used for the range bins between
                    the top and peak of bright band
phaseNearSurface = 175 is used for the range bins between
                    the peak and bottom of bright band

Integer values of phaseNearSurface/100 =
0 - solid
1 - mixed phase
2 - liquid
255 - Missing

epsilon (4-byte float, array size: nbin x nray x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution, 
$\epsilon = 1$ is no adjustment. Special values are defined as:
-9999.9 Missing value

FLG (Group in FS)

flagEcho (1-byte integer, array size: nbin x nray x nscan):

Flag of precipitation and main/side lobe clutter 
information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>

qualityData (4-byte integer, array size: nray x nscan):
Normal data gives "0". Non-zero values mean the kinds of errors. Special values are 
defined as:
-9999 Missing value

Flag of quality data. Bit range from 8 to 23 
contains flags by each module. Each module flag 
has 2 bits of information.

The 2 bit flag for each module has values:

[higher bit lower bit]
[0 0] Good
[0 1] Warning but usable
[1 0] NG or error

The bits of qualityData are assigned as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 7</td>
<td>Copy of dataQuality in level 1B product</td>
</tr>
<tr>
<td>8 - 9</td>
<td>Flag by input module</td>
</tr>
<tr>
<td>10 - 11</td>
<td>Flag by preparation module</td>
</tr>
<tr>
<td>12 - 13</td>
<td>Flag by vertical module</td>
</tr>
<tr>
<td>14 - 15</td>
<td>Flag by classification module</td>
</tr>
<tr>
<td>16 - 17</td>
<td>Flag by SRT module</td>
</tr>
<tr>
<td>18 - 19</td>
<td>Flag by DSD module</td>
</tr>
<tr>
<td>20 - 21</td>
<td>Flag by solver module</td>
</tr>
<tr>
<td>22 - 23</td>
<td>Flag by output module</td>
</tr>
<tr>
<td>24 - 31</td>
<td>Spare</td>
</tr>
</tbody>
</table>

**qualityFlag** (1-byte integer, array size: nray x nscan):
Flag derived from qualityData with the following values: Special values are defined as:

-99 Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

**flagSensor** (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>

**flagScanPattern** (2-byte integer, array size: nscan):
Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
**HS (Swath)**

**HS_SwathHeader (Metadata):**
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime (Group in HS)**
A UTC time associated with the scan.

- **Year** (2-byte integer, array size: nscan):
  4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
  
  -9999 Missing value

- **Month** (1-byte integer, array size: nscan):
  Month of the year. Values range from 1 to 12 months. Special values are defined as:
  
  -99 Missing value

- **DayOfMonth** (1-byte integer, array size: nscan):
  Day of the month. Values range from 1 to 31 days. Special values are defined as:
  
  -99 Missing value

- **Hour** (1-byte integer, array size: nscan):
  UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
  
  -99 Missing value

- **Minute** (1-byte integer, array size: nscan):
  Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
  
  -99 Missing value

- **Second** (1-byte integer, array size: nscan):
  Second of the minute. Values range from 0 to 60 s. Special values are defined as:
  
  -99 Missing value

- **MillisSecond** (2-byte integer, array size: nscan):
  Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
  
  -9999 Missing value

- **DayOfYear** (2-byte integer, array size: nscan):
  Day of the year. Values range from 1 to 366 days. Special values are defined as:
  
  -9999 Missing value

- **SecondOfDay** (8-byte float, array size: nscan):
  A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
  
  -9999.9 Missing value
Latitude (4-byte float, array size: nrayHS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
   -9999.9  Missing value
Longitude (4-byte float, array size: nrayHS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9  Missing value

scanStatus (Group in HS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

   Bit  Meaning if bit = 1
   0      missing
   5      geoError is not zero
   6      modeStatus is not zero

dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

   Bit  Meaning if bit = 1
   0      Beam matching is abnormal
   1      VPRF table is abnormal
   2      Surface table is abnormal
   3      geoWarning is not zero
   4      Operational mode is not observation mode
   5      GPS status is abnormal
   6      Spare (always 0)
   7      Check sum of L1A is abnormal

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

   Bit  Meaning if bit = 1
   0      Scan is missing
   1      Science telemetry packet missing
Science telemetry segment within packet missing
Science telemetry other missing
Housekeeping (HK) telemetry packet missing
Spare (always 0)
Spare (always 0)
Spare (always 0)

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

Bit Meaning if bit = 1
0 Spare (always 0)
1 SCorientation not 0 or 180
2 pointingStatus not 0
3 Non-routine limitErrorFlag
4 Non-routine operationalMode (not 1 or 11)
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)

goError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0 Latitude limit exceeded for viewed pixel locations
1 Negative scan time, invalid input
2 Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
Nadir calculation error for subsatellite position
Pixel count with geolocation error over threshold
Error in getting spacecraft attitude for any pixel
Error in getting spacecraft ephemeris for any pixel
Spare (always 0)
Spare (always 0)
Spare (always 0)
Spare (always 0)
Spare (always 0)
Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in
dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be
useful. A zero integer value indicates usual geolocation. A non-zero value broken down
into the following bit flags indicates the following, where bit 0 is the least significant bit
(i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion,
measured clockwise facing down. We define v in the same direction as the spacecraft
axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit
is set to 1 in modeStatus.
Value Meaning
0   +X forward (yaw 0)
180  -X forward (yaw 180)
-8000  Non-nominal pointing
-9999  Missing

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

Value Meaning
0   Nominal pointing in Mission Science Mode
1   GPS point solution stale and PVT ephemeris used
2   GEONS solution stale and GEONS ephemeris used
-8000  Non-nominal mission science orientation
-9999  Missing

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

Value Meaning
0   LAUNCH
1   RATENULL
2   SUNPOINT
3   GSPM (Gyro-less Sun Point)
4   MSM (Mission Science Mode)
5   SLEW
6   DELTAH
7   DELTAV
-99   UNKNOWN -- ACS mode unavailable

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

Value Meaning
0   S/C Z axis nadir, +X in flight direction
1   Flight Z axis nadir, +X in flight direction
2   S/C Z axis nadir, -X in flight direction
3   Flight Z axis nadir, -X in flight direction
4   +90 yaw for DPR antenna pattern calibration
5   -90 yaw for DPR antenna pattern calibration
-99   Missing
operationalMode (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks.
limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value
navigation (Group in HS)

scPos (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector (ms⁻¹) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees.
Special values are defined as:
-9999.9 Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0
to 100 s. Special values are defined as:
-9999.9  Missing value

**PRE (Group in HS)**

**elevation** (4-byte float, array size: nrayHS x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:
-9999.9  Missing value

**landSurfaceType** (4-byte integer, array size: nrayHS x nscan):
Land surface type.

0 - 99   Ocean
100 - 199 Land
200 - 299 Coast
300 - 399 Inland water
-9999    Missing value

**localZenithAngle** (4-byte float, array size: nrayHS x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values are in degree. Special values are defined as:
-9999.9  Missing value

**flagPrecip** (4-byte integer, array size: nrayHS x nscan):
Precipitation or no precipitation.

For L2 Ku and L2 Ka

0  No precipitation
1  Precipitation
-9999  Missing value

For L2 DPR

0  No precipitation by both Ku and Ka
1  Precipitation by Ka, no rain by Ku
10  Precipitation by Ku, no rain by Ka
11  Precipitation by both Ku and Ka
flagSigmaZeroSaturation (1-byte char, array size: nrayHS x nscan):

A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nrayHS x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nrayHS x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

heightStormTop (4-byte float, array size: nrayHS x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value

height (4-byte float, array size: nbinHS x nrayHS x nscan):
Height. Values are in m. Special values are defined as:
-9999.9 Missing value

binClutterFreeBottom (2-byte integer, array size: nrayHS x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

sigmaZeroMeasured (4-byte float, array size: nrayHS x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:
-9999.9 Missing value

zFactorMeasured (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values
are in dBZ. Special values are defined as:
-9999.9 Missing value

**ellipsoidBinOffset** (4-byte float, array size: nrayHS x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

\[
\text{ellipsoidBinOffset} = \text{scRangeEllipsoid} - \{ \text{startBinRange} + (\text{binEllipsoid}-1) \times \text{rangeBinSize} \}
\]

-9999 Missing value

**snRatioAtRealSurface** (4-byte float, array size: nrayHS x nscan):
Signal/Noise ratio at real surface range bin.

\[
\text{snRatioAtRealSurface} = 10.\times\log_{10}(\text{echoPowertrueV[mW]}/\text{noisePowertrueV[mW]})
\]

-9999 Missing value

**adjustFactor** (4-byte float, array size: nrayHS x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm’) and sigmaZeroMeasured (dBs0m’).

\[
\text{dBZm'} = \text{dBZm} - \text{adjustFactor} \\
\text{dBs0m'} = \text{dBs0m} - \text{adjustFactor}
\]

The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency, 
angle-bin adjustment for angle-bin dependency, and 
temporal adjustment for orbit number dependency.

**snowIceCover** (1-byte integer, array size: nrayHS x nscan):
TBD. Special values are defined as:
-99 Missing value

**VER** (Group in HS)
**airTemperature** (4-byte float, array size: nbinHS x nrayHS x nscan):
Air Temperature. Values are in K. Special values are defined as:
-9999.9 Missing value

**binZeroDeg** (2-byte integer, array size: nrayHS x nscan):
Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

**attenuationNP** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are defined as:
-9999.9 Missing value

**piaNP** (4-byte float, array size: nNP x nrayHS x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are defined as:
-9999.9 Missing value

**sigmaZeroNPCorrected** (4-byte float, array size: nrayHS x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

**heightZeroDeg** (4-byte float, array size: nrayHS x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9 Missing value

**CSF** (Group in HS)
**flagBB** (4-byte integer, array size: nrayHS x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.

**L2 DPR:**
0  no Bright Band
1  Bright Band detected by Ku and DFRm
2  Bright Band detected by Ku only
3  Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value

**L2 Ku and L2 Ka:**
0  BB not detected
1  BB detected
-1111 No rain value
-9999 Missing value

**binBBPeak** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binBBTop** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binBBBottom** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binHeavyIcePrecipTop** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the top of heavy ice precip. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with
88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binHeavyIcePrecipBottom** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the bottom of heavy ice precip. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**nHeavyIcePrecip** (1-byte char, array size: nrayHS x nscan):
TBD. Special values are defined as:
255 Missing value

**heightBB** (4-byte float, array size: nrayHS x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

**widthBB** (4-byte float, array size: nrayHS x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

**qualityBB** (4-byte integer, array size: nrayHS x nscan):

Quality of the bright band.
When the bright band is detected,
a larger positive number indicates lower confidence in the detection.

The Ku detection is clear, but the Ka and DPR detection is somewhat doubtful.

The meaning of qualityBB has not been finalized.

3 Smeared bright band
2 Not so clear bright band
1 Clear bright band
0 BB not detected in the case of rain
-1111 No rain value
-9999 Missing value
**typePrecip** (4-byte integer, array size: nrayHS x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, onvective, and other, can be obtained as follows:

When typePrecip is greater than zero,
Major rain type = typePrecip/10000000
  = 1    stratiform
  = 2    convective
  = 3    other

-1111  No rain value
-9999   Missing value

Let abcdefgh be the 8 digit number,

abcdefgh

then
  a: Main rain type. (a=1,2,3),
  b: 0,
  c: 0,
  d: V rain type,
  e: H rain type,
  f: BB,
  g: Shallow rain,
  h: Small size cell.

----------------------------------------------------
The following numbers appear as Ku and Ka (MS/HS) rain types:
---- stratiform
  1001H100
  10031000
---- convective
  2001H1xy (x>0 or y>0)
  2002Hbxy
  200310xy (x>0 or y>0)
  200320xy
---- other
  300330xy

where H is the rain type by H-method, and b depends on BB, x on shallow rain and y on small size cell:

H = 1: stratiform by H-method,
    2: convective by H-method,
    3: other by H-method.
\[ b = 0: \text{BB not detected}, \]
\[ 1: \text{BB detected}. \]
\[ x = 0: \text{No shallow rain}, \]
\[ 1: \text{Shallow isolated}, \]
\[ 3: \text{Shallow non-isolated}. \]
\[ y = 0: \text{No small size cell}, \]
\[ 1: \text{Single cell}, \]
\[ 2: \text{Small size cell consisting of two adjacent pixels}. \]

In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:

DFRm rain type = (typePrecip\%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000) in FORTRAN

DFRm rain type
\[ = 1 \quad \text{stratiform} \]
\[ = 2 \quad \text{convective} \]
\[ = 4 \quad \text{transition} \]
\[ = 8 \quad \text{DFRm method cannot be applicable at Part B (in this case the conventional method determines the major rain type)} \]
\[ = 9 \quad \text{DFRm method cannot be applicable at Part A (in this case the conventional method determines the major rain type)} \]

-1111 No rain value
-9999 Missing value

If dual frequency data is not available but Ku-only or Ka-only is available, rain type is expressed by the following 8 digit number:
\[ 10xxxxxx --- \text{stratiform}, \]
\[ 20xxxxxx --- \text{convective}, \]
\[ 30xxxxxx --- \text{other}, \]
which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by
\[ 1qxxxxxx --- \text{stratiform}, \]
\[ 2qxxxxxx --- \text{convective}, \]
\[ 3qxxxxxx --- \text{other}, \]
where \( q > 0 \).
Thus, by examining \( q \), users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.

For MS and HS, DFRm method is used.

DFRm decision classifies rain type into
- stratiform,
- convective,
and
- transition.

The DPR numbering rule can be summarized as follows:
Let \( \text{o p q r s t u v} \) be the 8 digit number, then
- \( \text{o} \): Main rain type. (\( \text{o}=1,2,3 \)),
- \( \text{p} \): DFRm rain type. (\( \text{p}=0,1,2,4,8,9 \), with \( \text{p}=0 \) for single frequency data only),
- \( \text{q} \): DFRm BB. (\( \text{q}=0,1 \)),
- \( \text{r} \): V rain type (by conventional V-method).
  - Basically \( r=0 \) for inner swath and \( r>0 \) for outer swath.
  - However, \( r>0 \) when only single frequency data is available,
- \( \text{s} \): H rain type,
- \( \text{t} \): = 0 for inner swath,
  - 1 when BB is detected in the outer swath.
- \( \text{u} \): Shallow rain,
- \( \text{v} \): Small size cell.

DFRm type can be obtained by examining \( p \)

The meaning of \( p \) is as follows:
- \( p = 0 \): single frequency data only (dual frequency data not available),
  - 1: stratiform by DFRm method,
  - 2: convective by DFRm method,
  - 4: transition by DFRm method,
  - 8: DFRm decision not available,
  - 9: DFRm decision not available.

Note that \( p>0 \) always in DPR processing, which is different from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, \( p=0 \) always.

The following numbers appear as DPR rain types:
** For NS outer swath **

--- stratiform
1901H100
19031000
--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy
--- other
390330xy

** For NS inner swath and MS **

--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
or NS rain >0 but no MS rain; NS V and H determine rain type
--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
(x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
(x>0 or y>0)
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type

***************
* For HS  *
***************
--- stratiform
11B0H000
14B01000
19001000 --- H decision only
--- convective
21B0H0x0 (x>0)
22B0H0x0
240010x0 (x>0, 24B010x0 with B=0)
240020x0
241010x0 (x>0, 24B010x0 with B=1)
290010x0 (x>0) --- H decision only
290020x0 --- H decision only
--- other
340030x0
390030x0 --- H decision only

where w depends on BB by conventional V-method, B on BB
by DFRm method, H on H-method, x on shallow rain
and y on small size cell:
  w = 0: BB not detected by conventional V-method,
        1: BB detected by conventional V-method.
  B = 0: BB not detected by DFRm method,
        1: BB detected by DFRm method.
  H = 1: stratiform by H-method,
        2: convective by H-method,
        3: other by H-method.
  x = 0: No shallow rain,
        1: Shallow isolated,
        3: Shallow non-isolated.
y = 0: No small size cell,
    1: Single cell,
    2: Small size cell consisting of two adjacent pixels.
In the above, x>0 and y>0 are taken care of in the function
R\_type\_classification\_dpr2().

qualityTypePrecip (4-byte integer, array size: nrayHS x nscan):

Quality of the precipitation type.

1   Good
-1111 No rain value
-9999 Missing value

flagShallowRain (4-byte integer, array size: nrayHS x nscan):

Type of shallow rain
0    No shallow rain
10   Shallow isolated (maybe)
11   Shallow isolated (certain)
20   Shallow non-isolated (maybe)
21   Shallow non-isolated (certain)
-1111 No rain value
-9999 Missing value

flagHeavyIcePrecip (1-byte integer, array size: nrayHS x nscan):
This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors
above the -10 degree C isotherm. Special values are defined as:
         -99 Missing value

SRT (Group in HS)

pathAtten (4-byte float, array size: nrayHS x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are
defined as:
    -9999.9 Missing value

PIAalt (4-byte float, array size: method x nrayHS x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The
path-integrated attenuation from the jth method, where
PIAalt (j=1) = PIA_Ku from forward along-track spatial at kth angle bin
PIAalt (j=2) = PIA_Ku from backward along-track spatial at kth angle bin
PIAalt (j=3) = PIA_Ku from forward hybrid at kth angle bin
PIAalt (j=4) = PIA_Ku from backward hybrid at kth angle bin
PIAalt (j=5) = PIA_Ku from temporal reference at kth angle bin
PIAalt (j=6) = PIA_Ku from light-rain temporal reference at kth angle bin
Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAhb (4-byte float, array size: nrayHS x nscan):
The 2-way attenuation of HB.
Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAhybrid (4-byte float, array size: nrayHS x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.
Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAweight (4-byte float, array size: method x nrayHS x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.
PIAweight_j = 1/sigma_j^2 * (1/Sum_j(1/sigma_j^2))
Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAweightHY (4-byte float, array size: two x nrayHS x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.
PIAweight_j = 1/sigma_j^2 * (1/Sum_j(1/sigma_j^2))
Values are in dB. Special values are defined as:
-9999.9 Missing value

refScanID (2-byte integer, array size: nearFar x foreBack x nrayHS x nscan):
The number of scan lines between the current scan and the beginning (or end) of the along-track reference data at each angle bin. The values are computed by the equation: Current Scan Number - Reference Scan Number. The values are positive for the Forward estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:
1,1 - Forward - Near reference
2,1 - Forward - Far reference
1,2 - Backward - Near reference
2,2 - Backward - Far reference

Special values are defined as:
-9999 Missing value

**reliabFactor** (4-byte float, array size: nrayHS x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
-9999.9 Missing value

**reliabFactorAlt** (4-byte float, array size: method x nrayHS x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAalt. Special values are defined as:
-9999.9 Missing value

**reliabFactorHY** (4-byte float, array size: nrayHS x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

**reliabFlag** (2-byte integer, array size: nrayHS x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:

\begin{align*}
= 1 & \text{ if } \text{Rel}_\text{eff} > 3 ; \text{PIAeff estimate is considered reliable} \\
= 2 & \text{ if } 3 \geq \text{Rel}_\text{eff} > 1 ; \text{PIAeff estimate is considered marginally reliable} \\
= 3 & \text{ if } \text{Rel}_\text{eff} \leq 1 ; \text{PIAeff is unreliable} \\
= 4 & \text{ if } \text{SNR at surface} < 2\text{dB}; \text{provides a lower bound to the path-attenuation} \\
= 9 & \text{ (no-rain case)}
\end{align*}

Special values are defined as:
-9999 Missing value

**reliabFlagHY** (2-byte integer, array size: nrayHS x nscan):
TBD.

Special values are defined as:
-9999 Missing value

**stddevEff** (4-byte float, array size: nsdew x nrayHS x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.
Special values are defined as:
-9999.9  Missing value

**stddevHY** (4-byte float, array size: nrayHS x nscan):
TBD.

Special values are defined as:
-9999.9  Missing value

**zeta** (4-byte float, array size: nrayHS x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
-9999.9  Missing value

**DSD** (Group in HS)

**phase** (1-byte char, array size: nbinHS x nrayHS x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

phase < 100  Temperature(C)=phase-100  
phase > 200  Temperature(C)=phase-200  
phase = 100  Top of the bright band  
phase = 200  Bottom of the bright band  
phase = 125  is used for the range bins between 
the top and peak of bright band  
phase = 175  is used for the range bins between 
the peak and bottom of bright band

Integer values of phase/100 =

0  -  solid  
1  -  mixed phase  
2  -  liquid  
255  -  Missing

**binNode** (2-byte integer, array size: nNode x nrayHS x nscan):
The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to
   binRealSurface in PRE group.

For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
-9999 - Missing

**Experimental** (Group in HS)

`precipRateESurface2` *(4-byte float, array size: nrayHS x nscan)*:
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

`precipRateESurface2Status` *(1-byte char, array size: nrayHS x nscan)*:
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
255 Missing value

`sigmaZeroProfile` *(4-byte float, array size: nbinSZPHS x nrayHS x nscan)*:
Surface backscattering cross section profile around the current ifov. For information on this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:
-9999.9 Missing value

`seaIceConcentration` *(4-byte float, array size: nrayHS x nscan)*:
Sea ice concentration estimated by Ku. For information on this experimental field contact
the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value

**SLV (Group in HS)**

**flagSLV** (1-byte integer, array size: nbinHS x nrayHS x nscan):
Special values are defined as:
-99 Missing value

**paramDSD** (4-byte float, array size: nDSD x nbinHS x nrayHS x nscan):
Parameters of the drop size distribution. The first index is dBNw; the second index is Dm in mm. Special values are defined as:
-9999.9 Missing value

**binEchoBottom** (2-byte integer, array size: nrayHS x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

**piaFinal** (4-byte float, array size: nrayHS x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

**sigmaZeroCorrected** (4-byte float, array size: nrayHS x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9 Missing value

**zFactorCorrected** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedESurface** (4-byte float, array size: nrayHS x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nrayHS x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value
**paramNUBF** (4-byte float, array size: nNUBF x nrayHS x nscan):
TBD. Special values are defined as:
-9999.9 Missing value

**precipRate** (4-byte float, array size: nbinHS x nrayHS x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipWaterIntegrated** (4-byte float, array size: LS x nrayHS x nscan):
Precipitation water vertically integrated. Values are in \(g/m^2\). Special values are defined as:
-9999.9 Missing value

**qualitySLV** (4-byte integer, array size: nrayHS x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are defined as:
-9999 Missing value

**precipRateNearSurface** (4-byte float, array size: nrayHS x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface** (4-byte float, array size: nrayHS x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateAve24** (4-byte float, array size: nrayHS x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**phaseNearSurface** (1-byte char, array size: nrayHS x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

phaseNearSurface < 100 Temperature(C)=phaseNearSurface-100
phaseNearSurface > 200 Temperature(C)=phaseNearSurface-200
phaseNearSurface = 100 Top of the bright band
phaseNearSurface = 200 Bottom of the bright band
phaseNearSurface = 125 is used for the range bins between
the top and peak of bright band
phaseNearSurface = 175 is used for the range bins between
the peak and bottom of bright band

Integer values of phaseNearSurface/100 =
0 - solid
1 - mixed phase
2 - liquid
255 - Missing

\textbf{epsilon} (4-byte float, array size: nbinHS x nrayHS x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution, 
\(\text{epsilon} = 1\) is no adjustment. Special values are defined as:
-9999.9 Missing value

\textbf{FLG} (Group in HS)

\textbf{flagEcho} (1-byte integer, array size: nbinHS x nrayHS x nscan):

Flag of precipitation and main/side lobe clutter information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>

\textbf{qualityData} (4-byte integer, array size: nrayHS x nscan):
Normal data gives "0". Non-zero values mean the kinds of errors. Special values are defined as:
-9999 Missing value

Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag has 2 bits of information.

The 2 bit flag for each module has values:
[higher bit lower bit]
[0 0] Good
The bits of qualityData are assigned as follows:

Bit  | Meaning
--- | ---
0 - 7 | Copy of dataQuality in level 1B product
8 - 9 | Flag by input module
10 - 11 | Flag by preparation module
12 - 13 | Flag by vertical module
14 - 15 | Flag by classification module
16 - 17 | Flag by SRT module
18 - 19 | Flag by DSD module
20 - 21 | Flag by solver module
22 - 23 | Flag by output module
24 - 31 | Spare

**qualityFlag** (1-byte integer, array size: nrayHS x nscan):
Flag derived from qualityData with the following values: Special values are defined as:
-99  Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

**flagSensor** (1-byte integer, array size: nscan):
Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>

**flagScanPattern** (2-byte integer, array size: nscan):
Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
C Structure Header file:

```c
#ifndef _TK_2AKaX_H_
define _TK_2AKaX_H_

#ifndef _L2AKaX_HS_FLG_
define _L2AKaX_HS_FLG_

typedef struct {
    signed char flagEcho[24][88];
    int qualityData[24];
    signed char qualityFlag[24];
    signed char flagSensor;
    short flagScanPattern;
} L2AKaX_HS_FLG;
#endif

define _L2AKaX_HS_SLV_
define _L2AKaX_HS_SLV_

typedef struct {
    signed char flagSLV[24][88];
    float paramDSD[24][88][2];
    short binEchoBottom[24];
    float piaFinal[24];
    float sigmaZeroCorrected[24];
    float zFactorCorrected[24][88];
    float zFactorCorrectedESurface[24];
    float zFactorCorrectedNearSurface[24];
    float paramNUBF[24][3];
    float precipRate[24][88];
    float precipWaterIntegrated[24][2];
    int qualitySLV[24];
    float precipRateNearSurface[24];
    float precipRateESurface[24];
    float precipRateAve24[24];
    unsigned char phaseNearSurface[24];
    float epsilon[24][88];
} L2AKaX_HS_SLV;
#endif
```
#ifndef _L2AKaX_HS_EXPERIMENTAL_
#define _L2AKaX_HS_EXPERIMENTAL_

typedef struct {
    float precipRateESurface2[24];
    unsigned char precipRateESurface2Status[24];
    float sigmaZeroProfile[24][5];
    float seaIceConcentration[24];
} L2AKaX_HS_EXPERIMENTAL;
#endif

#ifndef _L2AKaX_HS_DSD_
#define _L2AKaX_HS_DSD_

typedef struct {
    unsigned char phase[24][88];
    short binNode[24][5];
} L2AKaX_HS_DSD;
#endif

#ifndef _L2AKaX_HS_SRT_
#define _L2AKaX_HS_SRT_

typedef struct {
    float pathAtten[24];
    float PIAalt[24][6];
    float PIAhb[24];
    float PIAhybrid[24];
    float PIAweight[24][6];
    float PIAweightHY[24][2];
    short refScanID[24][2][2];
    float reliabFactor[24];
    float reliabFactorAlt[24][6];
    float reliabFactorHY[24];
    short reliabFlag[24];
    short reliabFlagHY[24];
    float stddevEff[24][3];
    float stddevHY[24];
    float zeta[24];
} L2AKaX_HS_SRT;
typedef struct {
    int flagBB[24];
    short binBBPeak[24];
    short binBBTop[24];
    short binBBBottom[24];
    short binHeavyIcePrecipTop[24];
    short binHeavyIcePrecipBottom[24];
    unsigned char nHeavyIcePrecip[24];
    float heightBB[24];
    float widthBB[24];
    int qualityBB[24];
    int typePrecip[24];
    int qualityTypePrecip[24];
    int flagShallowRain[24];
    signed char flagHeavyIcePrecip[24];
} L2AKaX_HS_CS;

#endif

#ifndef _L2AKaX_HS_VER_
#define _L2AKaX_HS_VER_

typedef struct {
    float airTemperature[24][88];
    short binZeroDeg[24];
    float attenuationNP[24][88];
    float piaNP[24][4];
    float sigmaZeroNPCorrected[24];
    float heightZeroDeg[24];
} L2AKaX_HS_VER;

#endif

#ifndef _L2AKaX_HS_PRE_
#define _L2AKaX_HS_PRE_

typedef struct {
    float elevation[24];

int landSurfaceType[24];
float localZenithAngle[24];
int flagPrecip[24];
unsigned char flagSigmaZeroSaturation[24];
short binRealSurface[24];
short binStormTop[24];
float heightStormTop[24];
float height[24][88];
short binClutterFreeBottom[24];
float sigmaZeroMeasured[24];
float zFactorMeasured[24][88];
float ellipsoidBinOffset[24];
float snRatioAtRealSurface[24];
float adjustFactor[24];
signed char snowIceCover[24];
} L2AKaX_HS_PRE;

#endif

#ifndef _L2AKaX_HS_SCANSTATUS_
#define _L2AKaX_HS_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2AKaX_HS_SCANSTATUS;

#endif

#ifndef _L2AKaX_HS_
#define _L2AKaX_HS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[24];
    float Longitude[24];
    L2AKaX_HS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2AKaX_HS_PRE PRE;
    L2AKaX_HS_VER VER;
    L2AKaX_HS_CSF CSF;
    L2AKaX_HS_SRT SRT;
    L2AKaX_HS_DSD DSD;
    L2AKaX_HS_EXPERIMENTAL Experimental;
    L2AKaX_HS_SLV SLV;
    L2AKaX_HS_FLG FLG;
} L2AKaX_HS;

#endif
#ifndef _L2AKaX_FS_FLG_
#define _L2AKaX_FS_FLG_

typedef struct {
    signed char flagEcho[49][176];
    int qualityData[49];
    signed char qualityFlag[49];
    signed char flagSensor;
    short flagScanPattern;
} L2AKaX_FS_FLG;

#endif
#ifndef _L2AKaX_FS_SLV_
#define _L2AKaX_FS_SLV_

typedef struct {
    signed char flagSLV[49][176];
    float paramDSD[49][176][2];
    short binEchoBottom[49];
    float piaFinal[49];
    float sigmaZeroCorrected[49];
    float zFactorCorrected[49][176];
    float zFactorCorrectedESurface[49];
    float zFactorCorrectedNearSurface[49];
} L2AKaX_FS_SLV;

#endif
float paramNUBF[49][3];
float precipRate[49][176];
float precipWaterIntegrated[49][2];
int qualitySLV[49];
float precipRateNearSurface[49];
float precipRateESurface[49];
float precipRateAve24[49];
unsigned char phaseNearSurface[49];
float epsilon[49][176];
}
L2AKaX_FS_SLV;

#endif

#ifndef _L2AKaX_FS_EXPERIMENTAL_
#define _L2AKaX_FS_EXPERIMENTAL_

typedef struct {
  float precipRateESurface2[49];
  unsigned char precipRateESurface2Status[49];
  float sigmaZeroProfile[49][7];
  float seaIceConcentration[49];
} L2AKaX_FS_EXPERIMENTAL;
#endif

#ifndef _L2AKaX_FS_DSD_
#define _L2AKaX_FS_DSD_

typedef struct {
  unsigned char phase[49][176];
  short binNode[49][5];
} L2AKaX_FS_DSD;
#endif

#ifndef _L2AKaX_FS_SRT_
#define _L2AKaX_FS_SRT_

typedef struct {
  float pathAtten[49];
  float PIAalt[49][6];
  float PIAhb[49];
  float PIAhybrid[49];
} L2AKaX_FS_SRT;
#endif
typedef struct {
  int flagBB[49];
  short binBBPeak[49];
  short binBBTop[49];
  short binBBBottom[49];
  short binHeavyIcePrecipTop[49];
  short binHeavyIcePrecipBottom[49];
  unsigned char nHeavyIcePrecip[49];
  float heightBB[49];
  float widthBB[49];
  int qualityBB[49];
  int typePrecip[49];
  int qualityTypePrecip[49];
  int flagShallowRain[49];
  signed char flagHeavyIcePrecip[49];
} L2AKaX_FS_CSF;

typedef struct {
  float airTemperature[49][176];
  short binZeroDeg[49];
float attenuationNP[49][176];
float piaNP[49][4];
float sigmaZeroNPCorrected[49];
float heightZeroDeg[49];
} L2AKaX_FS_VER;

#endif

#ifndef _L2AKaX_FS_PRE_
#define _L2AKaX_FS_PRE_

typedef struct {
  float elevation[49];
  int landSurfaceType[49];
  float localZenithAngle[49];
  int flagPrecip[49];
  unsigned char flagSigmaZeroSaturation[49];
  short binRealSurface[49];
  short binStormTop[49];
  float heightStormTop[49];
  float height[49][176];
  short binClutterFreeBottom[49];
  float sigmaZeroMeasured[49];
  float zFactorMeasured[49][176];
  float ellipsoidBinOffset[49];
  float snRatioAtRealSurface[49];
  float adjustFactor[49];
  signed char snowIceCover[49];
} L2AKaX_FS_PRE;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
  float scPos[3];
  float scVel[3];
  float scLat;
  float scLon;
  float scAlt;
  float dprAlt;
  float scAttRollGeoc;
} NAVIGATION;

#endif
float scAttPitchGeoc;
float scAttYawGeoc;
float scAttRollGeod;
float scAttPitchGeod;
float scAttYawGeod;
greenHourAng;
timeMidScan;
timeMidScanOffset;
} NAVIGATION;
#endif

#ifndef _L2AKaX_FS_SCANSTATUS_
#define _L2AKaX_FS_SCANSTATUS_

typedef struct {
signed char dataQuality;
signed char dataWarning;
signed char missing;
signed char modeStatus;
short geoError;
short geoWarning;
short SCorientation;
short pointingStatus;
signed char acsModeMidScan;
signed char targetSelectionMidScan;
signed char operationalMode;
signed char limitErrorFlag;
double FractionalGranuleNumber;
} L2AKaX_FS_SCANSTATUS;
#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
short Year;
signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
} SCANTIME;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L2AKaX_FS_
#define _L2AKaX_FS_

typedef struct {
   SCANTIME ScanTime;
   float Latitude[49];
   float Longitude[49];
   L2AKaX_FS_SCANSTATUS scanStatus;
   NAVIGATION navigation;
   L2AKaX_FS_PRE PRE;
   L2AKaX_FS_VER VER;
   L2AKaX_FS_CSF CSF;
   L2AKaX_FS_SRT SRT;
   L2AKaX_FS_DSD DSD;
   L2AKaX_FS_EXPERIMENTAL Experimental;
   L2AKaX_FS_SLV SLV;
   L2AKaX_FS_FLG FLG;
} L2AKaX_FS;
#endif

#endif

#ifndef _L2AKaX_SWATHS_
#define _L2AKaX_SWATHS_

typedef struct {
   L2AKaX_FS FS;
   L2AKaX_HS HS;
} L2AKaX_SWATHS;
#endif

#endif

Fortran Structure Header file:

STRUCTURE /L2AKaX_HS_FLG/
BYTE flagEcho(88,24)
INTEGER*4 qualityData(24)
BYTE qualityFlag(24)
BYTE flagSensor
INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2AKaX_HS_SLV/
BYTE flagSLV(88,24)
REAL*4 paramDSD(2,88,24)
INTEGER*2 binEchoBottom(24)
REAL*4 piaFinal(24)
REAL*4 sigmaZeroCorrected(24)
REAL*4 zFactorCorrected(88,24)
REAL*4 zFactorCorrectedESurface(24)
REAL*4 zFactorCorrectedNearSurface(24)
REAL*4 paramNUBF(3,24)
REAL*4 precipRate(88,24)
REAL*4 precipWaterIntegrated(2,24)
INTEGER*4 qualitySLV(24)
REAL*4 precipRateNearSurface(24)
REAL*4 precipRateESurface(24)
REAL*4 precipRateAve24(24)
CHARACTER phaseNearSurface(24)
REAL*4 epsilon(88,24)
END STRUCTURE

STRUCTURE /L2AKaX_HS_EXPERIMENTAL/
REAL*4 precipRateESurface2(24)
CHARACTER precipRateESurface2Status(24)
REAL*4 sigmaZeroProfile(5,24)
REAL*4 seaIceConcentration(24)
END STRUCTURE

STRUCTURE /L2AKaX_HS_DSD/
CHARACTER phase(88,24)
INTEGER*2 binNode(5,24)
END STRUCTURE

STRUCTURE /L2AKaX_HS_SRT/
REAL*4 pathAtten(24)
REAL*4 PIAalt(6,24)
REAL*4 PIAhb(24)
REAL*4 PIAhybrid(24)
REAL*4 PIAweight(6,24)
REAL*4 PIAweightHY(2,24)
INTEGER*2 refScanID(2,2,24)
REAL*4 reliabFactor(24)
REAL*4 reliabFactorAlt(6,24)
REAL*4 reliabFactorHY(24)
INTEGER*2 reliabFlag(24)
INTEGER*2 reliabFlagHY(24)
REAL*4 stddevEff(3,24)
REAL*4 stddevHY(24)
REAL*4 zeta(24)
END STRUCTURE

STRUCTURE /L2AKaX_HS_CSF/
  INTEGER*4 flagBB(24)
  INTEGER*2 binBBPeak(24)
  INTEGER*2 binBBTop(24)
  INTEGER*2 binBBBottom(24)
  INTEGER*2 binHeavyIcePrecipTop(24)
  INTEGER*2 binHeavyIcePrecipBottom(24)
  CHARACTER nHeavyIcePrecip(24)
  REAL*4 heightBB(24)
  REAL*4 widthBB(24)
  INTEGER*4 qualityBB(24)
  INTEGER*4 typePrecip(24)
  INTEGER*4 qualityTypePrecip(24)
  INTEGER*4 flagShallowRain(24)
  BYTE flagHeavyIcePrecip(24)
END STRUCTURE

STRUCTURE /L2AKaX_HS_VER/
  REAL*4 airTemperature(88,24)
  INTEGER*2 binZeroDeg(24)
  REAL*4 attenuationNP(88,24)
  REAL*4 piaNP(4,24)
  REAL*4 sigmaZeroNPCorrected(24)
  REAL*4 heightZeroDeg(24)
END STRUCTURE

STRUCTURE /L2AKaX_HS_PRE/
  REAL*4 elevation(24)
  INTEGER*4 landSurfaceType(24)
REAL*4 localZenithAngle(24)
INTEGER*4 flagPrecip(24)
CHARACTER flagSigmaZeroSaturation(24)
INTEGER*2 binRealSurface(24)
INTEGER*2 binStormTop(24)
REAL*4 heightStormTop(24)
REAL*4 height(88,24)
INTEGER*2 binClutterFreeBottom(24)
REAL*4 sigmaZeroMeasured(24)
REAL*4 zFactorMeasured(88,24)
REAL*4 ellipsoidBinOffset(24)
REAL*4 snRatioAtRealSurface(24)
REAL*4 adjustFactor(24)
BYTE snowIceCover(24)

END STRUCTURE

STRUCTURE /L2AKaX_HS_SCANSTATUS/
    BYTE dataQuality
    BYTE dataWarning
    BYTE missing
    BYTE modeStatus
    INTEGER*2 geoError
    INTEGER*2 geoWarning
    INTEGER*2 SCorientation
    INTEGER*2 pointingStatus
    BYTE acsModeMidScan
    BYTE targetSelectionMidScan
    BYTE operationalMode
    BYTE limitErrorFlag
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2AKaX_HS/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(24)
    REAL*4 Longitude(24)
    RECORD /L2AKaX_HS_SCANSTATUS/ scanStatus
    RECORD /NAVIGATION/ navigation
    RECORD /L2AKaX_HS_PRE/ PRE
    RECORD /L2AKaX_HS_VER/ VER
    RECORD /L2AKaX_HS_CSF/ CSF
    RECORD /L2AKaX_HS_SRT/ SRT
    RECORD /L2AKaX_HS_DSD/ DSD
RECORD /L2AKaX_HS_EXPERIMENTAL/ Experimental
RECORD /L2AKaX_HS_SLV/ SLV
RECORD /L2AKaX_HS_FLG/ FLG
END STRUCTURE

STRUCTURE /L2AKaX_FS_FLG/
  BYTE flagEcho(176,49)
  INTEGER*4 qualityData(49)
  BYTE qualityFlag(49)
  BYTE flagSensor
  INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2AKaX_FS_SLV/
  BYTE flagSLV(176,49)
  REAL*4 paramDSD(2,176,49)
  INTEGER*2 binEchoBottom(49)
  REAL*4 piaFinal(49)
  REAL*4 sigmaZeroCorrected(49)
  REAL*4 zFactorCorrected(176,49)
  REAL*4 zFactorCorrectedESurface(49)
  REAL*4 zFactorCorrectedNearSurface(49)
  REAL*4 paramNUBF(3,49)
  REAL*4 precipRate(176,49)
  REAL*4 precipWaterIntegrated(2,49)
  INTEGER*4 qualitySLV(49)
  REAL*4 precipRateNearSurface(49)
  REAL*4 precipRateESurface(49)
  REAL*4 precipRateAve24(49)
  CHARACTER phaseNearSurface(49)
  REAL*4 epsilon(176,49)
END STRUCTURE

STRUCTURE /L2AKaX_FS_EXPERIMENTAL/
  REAL*4 precipRateESurface2(49)
  CHARACTER precipRateESurface2Status(49)
  REAL*4 sigmaZeroProfile(7,49)
  REAL*4 seaIceConcentration(49)
END STRUCTURE

STRUCTURE /L2AKaX_FS_DSD/
  CHARACTER phase(176,49)
  INTEGER*2 binNode(5,49)
END STRUCTURE

STRUCTURE /L2AKaX_FS_SRT/
    REAL*4 pathAtten(49)
    REAL*4 PIAalt(6,49)
    REAL*4 PIAhb(49)
    REAL*4 PIAhybrid(49)
    REAL*4 PIAweight(6,49)
    REAL*4 PIAweightHY(2,49)
    INTEGER*2 refScanID(2,2,49)
    REAL*4 reliabFactor(49)
    REAL*4 reliabFactorAlt(6,49)
    REAL*4 reliabFactorHY(49)
    INTEGER*2 reliabFlag(49)
    INTEGER*2 reliabFlagHY(49)
    REAL*4 stddevEff(3,49)
    REAL*4 stddevHY(49)
    REAL*4 zeta(49)
END STRUCTURE

STRUCTURE /L2AKaX_FS_CSF/
    INTEGER*4 flagBB(49)
    INTEGER*2 binBBPeak(49)
    INTEGER*2 binBBTop(49)
    INTEGER*2 binBBBottom(49)
    INTEGER*2 binHeavyIcePrecipTop(49)
    INTEGER*2 binHeavyIcePrecipBottom(49)
    CHARACTER nHeavyIcePrecip(49)
    REAL*4 heightBB(49)
    REAL*4 widthBB(49)
    INTEGER*4 qualityBB(49)
    INTEGER*4 typePrecip(49)
    INTEGER*4 qualityTypePrecip(49)
    INTEGER*4 flagShallowRain(49)
    BYTE flagHeavyIcePrecip(49)
END STRUCTURE

STRUCTURE /L2AKaX_FS_VER/
    REAL*4 airTemperature(176,49)
    INTEGER*2 binZeroDeg(49)
    REAL*4 attenuationNP(176,49)
    REAL*4 piaNP(4,49)
    REAL*4 sigmaZeroNPCorrected(49)
REAL*4 heightZeroDeg(49)
END STRUCTURE

STRUCTURE /L2AKaX_FS_PRE/
  REAL*4 elevation(49)
  INTEGER*4 landSurfaceType(49)
  REAL*4 localZenithAngle(49)
  INTEGER*4 flagPrecip(49)
  CHARACTER flagSigmaZeroSaturation(49)
  INTEGER*2 binRealSurface(49)
  INTEGER*2 binStormTop(49)
  REAL*4 heightStormTop(49)
  REAL*4 height(176,49)
  INTEGER*2 binClutterFreeBottom(49)
  REAL*4 sigmaZeroMeasured(49)
  REAL*4 zFactorMeasured(176,49)
  REAL*4 ellipsoidBinOffset(49)
  REAL*4 snRatioAtRealSurface(49)
  REAL*4 adjustFactor(49)
  BYTE snowIceCover(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
  REAL*4 scPos(3)
  REAL*4 scVel(3)
  REAL*4 scLat
  REAL*4 scLon
  REAL*4 scAlt
  REAL*4 dprAlt
  REAL*4 scAttRollGeoc
  REAL*4 scAttPitchGeoc
  REAL*4 scAttYawGeoc
  REAL*4 scAttRollGeod
  REAL*4 scAttPitchGeod
  REAL*4 scAttYawGeod
  REAL*4 greenHourAng
  REAL*8 timeMidScan
  REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L2AKaX_FS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
BYTE missing
BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
BYTE limitErrorFlag
REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AKaX_FS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
  REAL*4 Longitude(49)
  RECORD /L2AKaX_FS_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L2AKaX_FS_PRE/ PRE
  RECORD /L2AKaX_FS_VER/ VER
  RECORD /L2AKaX_FS_CSFSF/ CSF
  RECORD /L2AKaX_FS_SRT/ SRT
  RECORD /L2AKaX_FS_DSD/ DSD
  RECORD /L2AKaX_FS_EXPERIMENTAL/ Experimental
  RECORD /L2AKaX_FS_SLV/ SLV
  RECORD /L2AKaX_FS_FLG/ FLG
END STRUCTURE

STRUCTURE /L2AKaX_SWATHS/
  RECORD /L2AKaX_FS/ FS;
5.62 2ADPRX - DPR precipitation

The DPR Level-2A product, 2ADPRX, ”DPR precipitation,” is written as a 2 swath structure. The swaths are FS, full scans, and HS, high sensitivity scans. The following sections describe the structure and contents of the format.

Dimension definitions:

- **nschan** var Number of scans in the granule.
- **nray** 49 Number of angle bins in each FS scan.
- **nraysHS** 24 Number of angle bins in each HS scan.
- **nbin** 176 Number of range bins in each FS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
- **nbinHS** 88 Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.
- **nfreq** 2 Number of frequencies.
- **nfreqHI** 3 Number of frequencies.
- **nbinSZP** 7 Number of range bins for sigmaZeroProfile.
- **nbinSZPHS** 5 Number of range bins for sigmaZeroProfile in each HS scan.
- **nNP** 4 Number of NP kinds.
- **nearFar** 2 Near reference, Far reference.
- **foreBack** 2 Forward, Backward.
- **method** 6 Number of SRT methods.
- **nsdew** 3 Number of standard deviation effective ways.
- **nNode** 5 Number of binNode.
- **nDSD** 2 Number of DSD parameters. Parameters are dBNw and Dm (mm).
- **LS** 2 Liquid, solid.
- **nNUBF** 3 Number of NUBF parameters.
- **two** 2 Number two.
- **three** 3 Number three.
- **thirty** 30 Number of NUBF parameters.
- **thirteen** 13 Number of NUBF parameters.
- **ten** 10 Number of NUBF parameters.
- **six** 6 Number of NUBF parameters.
- **four** 4 Number of NUBF parameters.
- **eight** 8 Number of NUBF parameters.

Figure 892 through Figure 918 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
### Figure 892: Data Format Structure for 2ADPRX, DPR precipitation

![Diagram of File structure]

- **File**
  - FileHeader
  - InputRecord
  - AlgorithmRuntimeInfo
  - NavigationRecord
  - FileInfo
  - JAXAInfo

#### FS
- **ScanTime** 19 bytes
- **Latitude** 4 bytes
- **Longitude** 4 bytes
- **scanStatus** 36 bytes
- **navigation** 84 bytes
- **PRE** 10671 bytes
- **VER** 105742 bytes
- **CSF** 2744 bytes
- **SRT** 9800 bytes
- **DSD** 9114 bytes
- **Experimental** 3577 bytes
- **SLV** 219079 bytes
- **FLG** 8924 bytes
- **TRG** 22393 bytes

#### HS

---

### Figure 893: Data Format Structure for 2ADPRX, FS
Figure 894: Data Format Structure for 2ADPRX, HS

Figure 895: Data Format Structure for 2ADPRX, FS, ScanTime
5.62 2ADPRX - DPR precipitation

Figure 896: Data Format Structure for 2ADPRX, FS, scanStatus

Figure 897: Data Format Structure for 2ADPRX, FS, navigation
Figure 898: Data Format Structure for 2ADPRX, FS, PRE

Figure 899: Data Format Structure for 2ADPRX, FS, VER
Figure 900: Data Format Structure for 2ADPRX, FS, CSF
Figure 901: Data Format Structure for 2ADPRX, FS, SRT

Figure 902: Data Format Structure for 2ADPRX, FS, DSD

Figure 903: Data Format Structure for 2ADPRX, FS, Experimental
Figure 904: Data Format Structure for 2ADPRX, FS, SLV

Figure 905: Data Format Structure for 2ADPRX, FS, FLG
Figure 906: Data Format Structure for 2ADPRX, TRG
5.62 2ADPRX - DPR precipitation

continued from last figure

NUBFcorrPIA 4 bytes Array: two x nray x nscan

triggerParameters 4 bytes Array: eight x nray x nscan

Figure 907: Data Format Structure for 2ADPRX, FS, TRG

ScanTime

Year 2 bytes Array: nscan
Month 1 byte Array: nscan
DayOfMonth 1 byte Array: nscan
Hour 1 byte Array: nscan
Minute 1 byte Array: nscan
Second 1 byte Array: nscan
MilliSecond 2 bytes Array: nscan
DayOfYear 2 bytes Array: nscan
SecondOfDay 8 bytes Array: nscan

Figure 908: Data Format Structure for 2ADPRX, HS, ScanTime

scanStatus

dataQuality 1 byte Array: nscan
dataWarning 1 byte Array: nscan
missing 1 byte Array: nscan
modeStatus 1 byte Array: nscan
geoError 2 bytes Array: nscan
geoWarning 2 bytes Array: nscan
SCorientation 2 bytes Array: nscan
pointingStatus 2 bytes Array: nscan
acsModeMidScan 1 byte Array: nscan
targetSelectionMidScan 1 byte Array: nscan
operationalMode 1 byte Array: nscan
limitErrorFlag 1 byte Array: nscan
FractionalGranuleNumber 8 bytes Array: nscan

Figure 909: Data Format Structure for 2ADPRX, HS, scanStatus
Figure 910: Data Format Structure for 2ADPRX, HS, navigation
Figure 911: Data Format Structure for 2ADPRX, HS, PRE

Figure 912: Data Format Structure for 2ADPRX, HS, VER
Figure 913: Data Format Structure for 2ADPRX, HS, CSF
Figure 914: Data Format Structure for 2ADPRX, HS, SRT
Figure 915: Data Format Structure for 2ADPRX, HS, DSD

Figure 916: Data Format Structure for 2ADPRX, HS, Experimental

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**FS** (Swath)

**FS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
Figure 917: Data Format Structure for 2ADPRX, HS, SLV

Figure 918: Data Format Structure for 2ADPRX, HS, FLG
ScanTime (Group in FS)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod.
Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in FS)

**dataQuality** (1-byte integer, array size: nfreq x nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$, the unsigned integer value is $2^i$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**dataWarning** (1-byte integer, array size: nfreq x nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nfreq x nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
**modeStatus** (1-byte integer, array size: nfreq x nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^{**i}$). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoError** (2-byte integer, array size: nfreq x nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^{**i}$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

**geoWarning** (2-byte integer, array size: nfreq x nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
pointingStatus (2-byte integer, array size: nfreq x nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is
good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit
in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System
telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nfreq x nscan):
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

**limitErrorFlag** (1-byte integer, array size: nfreq x nscan):

- Bit flags for every ray with information about echo power limit checks.
- limitErrorFlag may be used in modeStatus.
- Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):

The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:

-9999.9 Missing value

**navigation** (Group in FS)
scPos (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector (m/s) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:

-9999.9 Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:

-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:

-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:

-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 1000000000 s. Special values are defined as:

-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:

-9999.9 Missing value
**PRE** (Group in FS)

**elevation** (4-byte float, array size: nray x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:
-9999.9 Missing value

**landSurfaceType** (4-byte integer, array size: nray x nscan):
Land surface type.

0 - 99 Ocean
100 - 199 Land
200 - 299 Coast
300 - 399 Inland water
-9999 Missing value

**localZenithAngle** (4-byte float, array size: nfreq x nray x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values are in degree. Special values are defined as:
-9999.9 Missing value

**flagPrecip** (4-byte integer, array size: nray x nscan):
Precipitation or no precipitation.

For L2 Ku and L2 Ka

0 No precipitation
1 Precipitation
-9999 Missing value

For L2 DPR

0 No precipitation by both Ku and Ka
1 Precipitation by Ka, no rain by Ku
10 Precipitation by Ku, no rain by Ka
11 Precipitation by both Ku and Ka
-9999 Missing value
flagSigmaZeroSaturation (1-byte char, array size: nfreq x nray x nscan):

A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nfreq x nray x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nray x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

heightStormTop (4-byte float, array size: nray x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value

height (4-byte float, array size: nbin x nray x nscan):
Height. Values are in m. Special values are defined as:
-9999.9 Missing value

binClutterFreeBottom (2-byte integer, array size: nray x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

sigmaZeroMeasured (4-byte float, array size: nfreq x nray x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:
-9999.9 Missing value

zFactorMeasured (4-byte float, array size: nfreq x nbin x nray x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:
-9999.9 Missing value
**ellipsoidBinOffset** (4-byte float, array size: nfreq x nray x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

\[
\text{ellipsoidBinOffset} = \text{scRangeEllipsoid} - (\text{startBinRange} + (\text{binEllipsoid}-1) \times \text{rangeBinSize})
\]

- **scRangeEllipsoid**: Distance between a sensor and the ellipsoid [m]
- **startBinRange**: Distance between a sensor and a center of the highest observed range bin [m]
- **binEllipsoid**: Range bin number of the Ellipsoid (1 - 260)
- **rangeBinSize**: Range bin size [m]

-9999 Missing value

**snRatioAtRealSurface** (4-byte float, array size: nfreq x nray x nscan):
Signal/Noise ratio at real surface range bin.

\[
\text{snRatioAtRealSurface} = 10. \times \log_{10}(\text{echoPowertrueV[mW]}/\text{noisePowertrueV[mW]})
\]

-9999 Missing value

**adjustFactor** (4-byte float, array size: nfreq x nray x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm') and sigmaZeroMeasured (dBs0m').

dBZm' = dBZm - adjustFactor
dBs0m' = dBs0m - adjustFactor

The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
temporal adjustment for orbit number dependency.

**snowIceCover** (1-byte integer, array size: nray x nscan):
TBD. Special values are defined as:
-99 Missing value

**VER** (Group in FS)

**airTemperature** (4-byte float, array size: nbin x nray x nscan):
Air Temperature. Values are in K. Special values are defined as:
-9999.9 Missing value
binZeroDeg (2-byte integer, array size: nray x nscan):

Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

attenuationNP (4-byte float, array size: nfreq x nbin x nray x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB/km. Special values are defined as:
-9999.9 Missing value

piaNP (4-byte float, array size: nfreq x nNP x nray x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water, cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are defined as:
-9999.9 Missing value

sigmaZeroNPCorrected (4-byte float, array size: nfreq x nray x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

heightZeroDeg (4-byte float, array size: nray x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9 Missing value

CSF (Group in FS)

flagBB (4-byte integer, array size: nray x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.
0    no Bright Band
1    Bright Band detected by Ku and DFRm
2    Bright Band detected by Ku only
3    Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value

L2 Ku and L2 Ka:
0    BB not detected
1    BB detected
-1111 No rain value
-9999 Missing value

**binBBPeak** (2-byte integer, array size: nray x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are
1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS
swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88
at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are
defined as:
-9999 Missing value

**binBBTop** (2-byte integer, array size: nray x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are
1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS
swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88
at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are
defined as:
-9999 Missing value

**binDFRmMLBottom** (2-byte integer, array size: nray x nscan):
Range bin number for melting layer bottom detected by the DFRm method.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;0</td>
<td>Range bin number when ML bottom is detected</td>
</tr>
<tr>
<td>0</td>
<td>ML bottom not detected</td>
</tr>
<tr>
<td>-1111</td>
<td>Value for no rain in MS(HS) mode at Ka band</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**binDFRmMLTop** (2-byte integer, array size: nray x nscan):
Range bin number for melting layer top detected by the DFRm method.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;0</td>
<td>Range bin number when ML top is detected</td>
</tr>
<tr>
<td>0</td>
<td>ML top not detected</td>
</tr>
<tr>
<td>-1111</td>
<td>Value for no rain in MS(HS) mode at Ka band</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
binBBBottom (2-byte integer, array size: nray x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
  -9999  Missing value

binHeavyIcePrecipTop (2-byte integer, array size: nfreqHI x nray x nscan):
Range bin number for the top of heavy ice precip. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
  -9999  Missing value

binHeavyIcePrecipBottom (2-byte integer, array size: nfreqHI x nray x nscan):
Range bin number for the bottom of heavy ice precip. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
  -9999  Missing value

nHeavyIcePrecip (1-byte char, array size: nfreqHI x nray x nscan):
The number of heavy ice precip. Special values are defined as:
  255  Missing value

flagMLquality (1-byte char, array size: nray x nscan):
ML quality flag. Special values are defined as:
  255  Missing value

heightBB (4-byte float, array size: nray x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
  -9999.9  Missing value

widthBB (4-byte float, array size: nray x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
  -9999.9  Missing value

qualityBB (4-byte integer, array size: nray x nscan):

Quality of the bright band.
When the bright band is detected,
a larger positive number indicates lower confidence in the detection.
The Ku detection is clear, but
the Ka and DPR detection is
somewhat doubtful.

The meaning of qualityBB has not
been finalized.

3  Smeared bright band
2  Not so clear bright band
1  Clear bright band
0  BB not detected in the case of rain
-1111 No rain value
-9999 Missing value

typePrecip (4-byte integer, array size: nray x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories,
stratiform, onvective, and other, can be obtained as follows:

When typePrecip is greater than zero,
Major rain type = typePrecip/10000000
  = 1  stratiform
  = 2  convective
  = 3  other

-1111 No rain value
-9999 Missing value

Let abcdefgh be the 8 digit number,

abcdefgh

then
  a: Main rain type. (a=1,2,3),
  b: 0,
  c: 0,
  d: V rain type,
  e: H rain type,
  f: BB,
  g: Shallow rain,
  h: Small size cell.

------------------------------------------------------------------
The following numbers appear as Ku and Ka (MS/HS) rain types:
  ---- stratiform
where H is the rain type by H-method, and b depends on BB,
x on shallow rain and y on small size cell:

H = 1: stratiform by H-method,
    2: convective by H-method,
    3: other by H-method.

b = 0: BB not detected,
    1: BB detected.

x = 0: No shallow rain,
    1: Shallow isolated,
    3: Shallow non-isolated.

y = 0: No small size cell,
    1: Single cell,
    2: Small size cell consisting of two adjacent pixels.

In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:

DFRm rain type = (typePrecip%10000000)/1000000 in C
DFRm rain type = (MOD(typePrecip,10000000)/1000000) in FORTRAN

DFRm rain type
    = 1  stratiform
    = 2  convective
    = 4  transition
    = 8  DFRm method cannot be applicable at Part B (in this case
         the conventional method determines the major rain type)
    = 9  DFRm method cannot be applicable at Part A (in this case
         the conventional method determines the major rain type)

-1111  No rain value
-9999   Missing value
If dual frequency data is not available but Ku-only or Ka-only is available, rain type is expressed by the following 8 digit number:

- 10xxxxxx --- stratiform,
- 20xxxxxx --- convective,
- 30xxxxxx --- other,

which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by

- 1qxxxxxx --- stratiform,
- 2qxxxxxx --- convective,
- 3qxxxxxx --- other,

where q>0.

Thus, by examining q, users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.

For MS and HS, DFRm method is used.

DFRm decision classifies rain type into

- stratiform,
- convective,

and

transition.

The DPR numbering rule can be summarized as follows:

Let opqrstuv be the 8 digit number, then

- o: Main rain type. (o=1,2,3),
- p: DFRm rain type. (p=0,1,2,4,8,9, with p=0 for single frequency data only),
- q: DFRm BB. (q=0,1),
- r: V rain type (by conventional V-method).
  - Basically r=0 for inner swath and r>0 for outer swath.
  - However, r>0 when only single frequency data is available,
- s: H rain type,
- t: = 0 for inner swath,
  - 1 when BB is detected in the outer swath.
- u: Shallow rain,
- v: Small size cell.

DFRm type can be obtained by examining p
The meaning of p is as follows:
p = 0: single frequency data only (dual frequency data not available),
  1: stratiform by DFRm method,
  2: convective by DFRm method,
  4: transition by DFRm method,
  8: DFRm decision not available,
  9: DFRm decision not available.
Note that p>0 always in DPR processing, which is different from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, p=0 always.

The following numbers appear as DPR rain types:

* For NS outer swath *

--- stratiform
1901H100
19031000
--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy
--- other
390330xy

* For NS inner swath and MS *

--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
  or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
  or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
  or NS rain >0 but no MS rain; NS V and H determine rain type
--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
(x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
(x>0 or y>0)
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type

**************
* For HS *
**************
--- stratiform
11B0H000
14B01000
19001000 --- H decision only
--- convective
21B0H0x0 (x>0)
22B0H0x0
240010x0 (x>0, 24B010x0 with B=0)
240020x0
241010x0 (x>0, 24B010x0 with B=1)
290010x0 (x>0) --- H decision only
290020x0 --- H decision only
--- other
340030x0
390030x0 --- H decision only

where w depends on BB by conventional V-method, B on BB
by DFRm method, H on H-method, x on shallow rain
and y on small size cell:

\( \text{w} = 0: \text{BB not detected by conventional V-method,} \)
\( \text{1: BB detected by conventional V-method}. \)

\( \text{B} = 0: \text{BB not detected by DFRm method,} \)
\( \text{1: BB detected by DFRm method}. \)

\( \text{H} = 1: \text{stratiform by H-method,} \)
\( \text{2: convective by H-method,} \)
\( \text{3: other by H-method}. \)

\( \text{x} = 0: \text{No shallow rain,} \)
\( \text{1: Shallow isolated,} \)
\( \text{3: Shallow non-isolated}. \)

\( \text{y} = 0: \text{No small size cell,} \)
\( \text{1: Single cell,} \)
\( \text{2: Small size cell consisting of two adjacent pixels}. \)

In the above, \( x>0 \) and \( y>0 \) are taken care of in the function
\( R\textunderscore\text{type}\textunderscore\text{classification}\textunderscore\text{dpr2}(). \)

\begin{verbatim}
 qualityTypePrecip (4-byte integer, array size: nray x nscan):

 Quality of the precipitation type.

  1     Good
 -1111  No rain value
 -9999  Missing value

 flagShallowRain (4-byte integer, array size: nray x nscan):

 Type of shallow rain
  0   No shallow rain
 10   Shallow isolated (maybe)
 11   Shallow isolated (certain)
 20   Shallow non-isolated (maybe)
 21   Shallow non-isolated (certain)
-1111  No rain value
-9999  Missing value
\end{verbatim}
flagHeavyIcePrecip (1-byte integer, array size: nray x nscan):
This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:
-99 Missing value

flagAnvil (1-byte integer, array size: nray x nscan):
flagAnvil is 1 when anvil is detected by the Ku-band radar,
0 when anvil is not detected, and
-99 when the data is missing.

Note that Ka-band decision is not made because of a lower sensitivity of Ka-band radar (therefore, there does not exist any Ka-band flagAnvil; only Ku-band flagAnvil is available in Ku-only and DPR NS).

SRT (Group in FS)

pathAtten (4-byte float, array size: nfreq x nray x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAalt (4-byte float, array size: nfreq x method x nray x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where
PIAalt (j=1) = PIA_Ku from forward along-track spatial at kth angle bin
PIAalt (j=2) = PIA_Ku from backward along-track spatial at kth angle bin
PIAalt (j=3) = PIA_Ku from forward hybrid at kth angle bin
PIAalt (j=4) = PIA_Ku from backward hybrid at kth angle bin
PIAalt (j=5) = PIA_Ku from temporal reference at kth angle bin
PIAalt (j=6) = PIA_Ku from light-rain temporal reference at kth angle bin
Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAdw (4-byte float, array size: nfreq x nray x nscan):
The 2-way attenuation.
Values are in dB. Special values are defined as:
-9999.9 Missing value
PIAhb (4-byte float, array size: nfreq x nray x nscan):
The 2-way attenuation of HB.

Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAhybrid (4-byte float, array size: nfreq x nray x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.

Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAweight (4-byte float, array size: method x nray x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

PIAweight_j = 1/sigma_j^2 * (1/Sum_j(1/sigma_j^2))

Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAweightHY (4-byte float, array size: three x nray x nscan):
The weights of the individual PIA_Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

PIAweight_j = 1/sigma_j^2 * (1/Sum_j(1/sigma_j^2))

Values are in dB. Special values are defined as:
-9999.9 Missing value

refScanID (2-byte integer, array size: nearFar x foreBack x nray x nscan):
The number of scan lines between the current scan and the beginning (or end) of the along-track reference data at each angle bin. The values are computed by the equation: Current Scan Number - Reference Scan Number. The values are positive for the Forward estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:

1,1 - Forward - Near reference
2,1 - Forward - Far reference
1,2 - Backward - Near reference
2,2 - Backward - Far reference
Special values are defined as:
-9999  Missing value

`reliabFactor` (4-byte float, array size: nray x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
-9999.9  Missing value

`reliabFactorAlt` (4-byte float, array size: method x nray x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAAlt. Special values are defined as:
-9999.9  Missing value

`reliabFactorHY` (4-byte float, array size: nray x nscan):
TBD.

Special values are defined as:
-9999.9  Missing value

`reliabFlag` (2-byte integer, array size: nray x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:
= 1 if Rel_eff > 3 ; PIA_eff estimate is considered reliable
= 2 if 3 ≥ Rel_eff > 1 ; PIA_eff estimate is considered marginally reliable
= 3 if Rel_eff ≤ 1 ; PIA_eff is unreliable
= 4 if SNR_at surface < 2dB; provides a lower bound to the path-attenuation
= 9 (no-rain case)

Special values are defined as:
-9999  Missing value

`reliabFlagHY` (2-byte integer, array size: nray x nscan):
TBD.

Special values are defined as:
-9999  Missing value

`stddevEff` (4-byte float, array size: nfreq x nsdew x nray x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.
Special values are defined as:
-9999.9 Missing value

\texttt{stddevHY} (4-byte float, array size: nfreq x nray x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

\texttt{zeta} (4-byte float, array size: nfreq x nray x nscan):
The term in the HB estimate of path attenuation.

Special values are defined as:
-9999.9 Missing value

\textbf{DSD} (Group in FS)

\texttt{phase} (1-byte char, array size: nbin x nray x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

\begin{itemize}
  \item phase < 100 \quad \text{Temperature(C)=phase-100}
  \item phase > 200 \quad \text{Temperature(C)=phase-200}
  \item phase = 100 \quad \text{Top of the bright band}
  \item phase = 200 \quad \text{Bottom of the bright band}
  \item phase = 125 \quad \text{is used for the range bins between the top and peak of bright band}
  \item phase = 175 \quad \text{is used for the range bins between the peak and bottom of bright band}
\end{itemize}

Integer values of phase/100 =

\begin{itemize}
  \item 0 - solid
  \item 1 - mixed phase
  \item 2 - liquid
  \item 255 - Missing
\end{itemize}

\texttt{binNode} (2-byte integer, array size: nNode x nray x nscan):
The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to
   binRealSurface in PRE group.

For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
-9999 - Missing

**Experimental** (Group in FS)

**precipRateESurface2** (4-byte float, array size: nray x nscan):
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
- -9999.9 Missing value

**precipRateESurface2Status** (1-byte char, array size: nray x nscan):
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
- 255 Missing value

**sigmaZeroProfile** (4-byte float, array size: nfreq x nbinSZP x nray x nscan):
Surface backscattering cross section profile around the current ifov. For information on this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:
- -9999.9 Missing value

**seaIceConcentration** (4-byte float, array size: nray x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact
the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value

flagSurfaceSnowfall (1-byte char, array size: nray x nscan):
Flag indicating snowfall on the surface, not aloft. 1 for snow, 0 for not snow. Special values are defined as:
255 Missing value

flagGraupelHail (1-byte char, array size: nray x nscan):
Graupel or Hail flag. Special values are defined as:
255 Missing value

binMixedPhaseTop (2-byte integer, array size: nray x nscan):
The range bin of the mixed phase top. Special values are defined as:
-9999 Missing value

surfaceSnowfallIndex (4-byte float, array size: nray x nscan):
Housekeeping product for test purposes. Special values are defined as:
-9999.9 Missing value

SLV (Group in FS)

flagSLV (1-byte integer, array size: nbin x nray x nscan):
Special values are defined as:
-99 Missing value

paramDSD (4-byte float, array size: nDSD x nbin x nray x nscan):
Parameters of the drop size distribution. The first index is dBNw; the second index is Dm in mm. Special values are defined as:
-9999.9 Missing value

binEchoBottom (2-byte integer, array size: nray x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

piaFinal (4-byte float, array size: nfreq x nray x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

sigmaZeroCorrected (4-byte float, array size: nfreq x nray x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9 Missing value
**zFactorCorrected** (4-byte float, array size: nfreq x nbin x nray x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedESurface** (4-byte float, array size: nfreq x nray x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nfreq x nray x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**paramNUBF** (4-byte float, array size: nNUBF x nray x nscan):
TBD. Special values are defined as:
-9999.9 Missing value

**precipRate** (4-byte float, array size: nbin x nray x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipWaterIntegrated** (4-byte float, array size: LS x nray x nscan):
Precipitation water vertically integrated. Values are in $g/m^2$. Special values are defined as:
-9999.9 Missing value

**qualitySLV** (4-byte integer, array size: nray x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are defined as:
-9999 Missing value

**precipRateNearSurface** (4-byte float, array size: nray x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface** (4-byte float, array size: nray x nscan):
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateAve24** (4-byte float, array size: nray x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**phaseNearSurface** (1-byte char, array size: nray x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in
the DSD group at the Near-surface level. As an unsigned byte value this represents:

\[
\begin{align*}
\text{phaseNearSurface} < 100 & \quad \text{Temperature}({}^\circ\text{C}) = \text{phaseNearSurface} - 100 \\
\text{phaseNearSurface} > 200 & \quad \text{Temperature}({}^\circ\text{C}) = \text{phaseNearSurface} - 200 \\
\text{phaseNearSurface} = 100 & \quad \text{Top of the bright band} \\
\text{phaseNearSurface} = 200 & \quad \text{Bottom of the bright band} \\
\text{phaseNearSurface} = 125 & \quad \text{is used for the range bins between} \\
& \quad \text{the top and peak of bright band} \\
\text{phaseNearSurface} = 175 & \quad \text{is used for the range bins between} \\
& \quad \text{the peak and bottom of bright band}
\end{align*}
\]

Integer values of \(\text{phaseNearSurface}/100 =\)

- 0 - solid \\
- 1 - mixed phase \\
- 2 - liquid \\
- 255 - Missing

**epsilon** (4-byte float, array size: nbin x nray x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution, \(\epsilon = 1\) is no adjustment. Special values are defined as:
- -9999.9 - Missing value

**FLG** (Group in FS)

**flagEcho** (1-byte integer, array size: nbin x nray x nscan):
Flag of precipitation and main/side lobe clutter information of each range bin.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)</td>
</tr>
<tr>
<td>0</td>
<td>For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)</td>
</tr>
<tr>
<td>1</td>
<td>Precipitation judged by L2 DPR algorithm</td>
</tr>
<tr>
<td>2</td>
<td>Precipitation judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>3</td>
<td>Precipitation judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>4</td>
<td>Main lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>5</td>
<td>Main lobe clutter judged by L2 Ka algorithm</td>
</tr>
<tr>
<td>6</td>
<td>Side lobe clutter judged by L2 Ku algorithm</td>
</tr>
<tr>
<td>7</td>
<td>Side lobe clutter judged by L2 Ka algorithm</td>
</tr>
</tbody>
</table>
**qualityData** (4-byte integer, array size: nray x nscan):
Normal data gives "0". Non-zero values mean the kinds of errors. Special values are defined as:
-9999  Missing value

Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag has 2 bits of information.

The 2 bit flag for each module has values:

- [0 0] Good
- [0 1] Warning but usable
- [1 0] NG or error

The bits of qualityData are assigned as follows:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 7</td>
<td>Copy of dataQuality in level 1B product</td>
</tr>
<tr>
<td>8 - 9</td>
<td>Flag by input module</td>
</tr>
<tr>
<td>10 - 11</td>
<td>Flag by preparation module</td>
</tr>
<tr>
<td>12 - 13</td>
<td>Flag by vertical module</td>
</tr>
<tr>
<td>14 - 15</td>
<td>Flag by classification module</td>
</tr>
<tr>
<td>16 - 17</td>
<td>Flag by SRT module</td>
</tr>
<tr>
<td>18 - 19</td>
<td>Flag by DSD module</td>
</tr>
<tr>
<td>20 - 21</td>
<td>Flag by solver module</td>
</tr>
<tr>
<td>22 - 23</td>
<td>Flag by output module</td>
</tr>
<tr>
<td>24 - 31</td>
<td>Spare</td>
</tr>
</tbody>
</table>

**qualityFlag** (1-byte integer, array size: nfreq x nray x nscan):
Flag derived from qualityData with the following values: Special values are defined as:
-99  Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

**flagSensor** (1-byte integer, array size: nfreq x nscan):

Flag of input Ku/Ka data condition.
<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>

**flagScanPattern** (2-byte integer, array size: nfreq x nscan):

Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**TRG** (Group in FS)
This is an experimental part of the retrieval algorithm. Currently all fields within this group are set to zero.

**NUBFIndex** (4-byte float, array size: nray x nscan):

Trigger Primary Output: final index of NUBF presence.
Integer between 0 and 100.
This field currently set to all zero.

**MSindex** (1-byte char, array size: nray x nscan):

Trigger Primary Output: final index of MS presence.
Float between 0 and 100.
This field currently set to all zero.

**MSindexKu** (1-byte integer, array size: nray x nscan):

Trigger Primary Output: final index of MS presence at Ku.
Integer between 0 and 100.
This field currently set to all zero.

**MSindexKa** (1-byte integer, array size: nray x nscan):

Trigger Primary Output: final index of MS presence at Ka.
Integer between 0 and 100.
This field currently set to all zero.
**precipFrac** (1-byte char, array size: three x nray x nscan):

Trigger Primary Output: number of neighbors estimated to be "empty" in the 3 neighborhoods (4MS, 4MS+4HS, 8MS+4HS)
This field currently set to all zero.

**RNUBFcond** (4-byte float, array size: nray x nscan):

Trigger Primary Output: estimate of Sigma n (as defined in Iguchi et al. 2000)
This field currently set to all zero.

**MSsurfPeakIndexKu** (1-byte char, array size: nray x nscan):

Trigger Secondary Output: index of surface peak reliability for the purpose of MS assessment at Ku.
This field currently set to all zero.

**MSsurfPeakIndexKa** (1-byte char, array size: nray x nscan):

Trigger Secondary Output: index of surface peak reliability for the purpose of MS assessment at Ka.
This field currently set to all zero.

**MSthroughsurfIndexKu** (1-byte char, array size: nray x nscan):

Trigger Secondary Output: index of MS tail through surface at Ku.
This field currently set to all zero.

**MSthroughsurfIndexKa** (1-byte char, array size: nray x nscan):

Trigger Secondary Output: index of MS tail through surface at Ka.
This field currently set to all zero.

**MSkneeDFRindex** (1-byte char, array size: nray x nscan):
Trigger Secondary Output: index of DFR Knee presence.
This field currently set to all zero.

**MSthrZindex** (1-byte char, array size: nray x nscan):

Trigger Secondary Output: high Z in ice index.
This field currently set to all zero.

**NUMFratioPIAindex** (1-byte char, array size: nray x nscan):

This field currently set to all zero.

**NUBFnZmVarIndex** (1-byte char, array size: three x nray x nscan):

Trigger Secondary Output: NUBF index based on the variability of Z (flat weight) in 4/8/12 neighbors at Ka
This field currently set to all zero.

**NUBFnZkVarIndex** (1-byte char, array size: three x nray x nscan):

Trigger Secondary Output: NUBF index based on the variability of Z (k-weighted) in 4/8/12 neighbors at Ka
This field currently set to all zero.

**NUBFnZmVarScaling** (2-byte integer, array size: nray x nscan):

Trigger Secondary Output: scaling of the NUBFnZmVarIndex
This field currently set to all zero.

**NUBFnZkVarScaling** (2-byte integer, array size: nray x nscan):

Trigger Secondary Output: scaling of the NUBFnZkVarIndex
This field currently set to all zero.

**NUBFsurfSliceIndex** (4-byte float, array size: thirty x nray x nscan):
Placeholder for the Surface Range Slicing Approach by Meneghini and Liang.
This field currently set to all zero.

**NUBFprofZPC** (4-byte float, array size: thirty x nray x nscan):

Placeholder for the Z PC approach by Haddad.
This field currently set to all zero.

**MSbreakpoints** (2-byte integer, array size: thirteen x nray x nscan):

Trigger diagnostic. 3 range bins selected for the Knee check, and 5 for the through Surface check (for each Ku and Ka).
This field currently set to all zero.

**MSslopes** (4-byte float, array size: ten x nray x nscan):

Trigger diagnostic. 2 slopes for the Knee check, and 4 for the through Surface check.
This field currently set to all zero.

**MSslopePoints** (4-byte float, array size: thirteen x nray x nscan):

Trigger diagnostic. Zfit values at 13 critical breakpoints.
This field currently set to all zero.

**MSslopeFits** (4-byte float, array size: six x nray x nscan):

Trigger diagnostic. Rmse for the 5 slope fits.
This field currently set to all zero.

**MSlowSNRrangeFilter** (1-byte char, array size: four x nray x nscan):

Trigger diagnostic. Type and length of the 2 filters used to regularize profile below SNR.
This field currently set to all zero.

**NUBFcorrPIA** (4-byte float, array size: two x nray x nscan):
Trigger diagnostic. Final PIA after reconciliation, used for the NUBRATIOPIAindex.
This field currently set to all zero.

triggerParameters (4-byte float, array size: eight x nray x nscan):

Trigger configuration. Set of tunable parameters (not output of the algorithm).
Only for version control.
This field currently set to all zero.

HS (Swath)

HS_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in HS)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value
MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nrayHS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nrayHS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in HS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
</tbody>
</table>
4 Operational mode is not observation mode
5 GPS status is abnormal
6 Spare (always 0)
7 Check sum of L1A is abnormal

missing (1-byte integer, array size: nscan): Indicates whether information is contained in the scan data. The values are:

   Bit  Meaning if bit = 1
   0   Scan is missing
   1   Science telemetry packet missing
   2   Science telemetry segment within packet missing
   3   Science telemetry other missing
   4   Housekeeping (HK) telemetry packet missing
   5   Spare (always 0)
   6   Spare (always 0)
   7   Spare (always 0)

modeStatus (1-byte integer, array size: nscan): A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

   Bit  Meaning if bit = 1
   0   Spare (always 0)
   1   SCoffsetion not 0 or 180
   2   pointingStatus not 0
   3   Non-routine limitErrorFlag
   4   Non-routine operationalMode (not 1 or 11)
   5   Spare (always 0)
   6   Spare (always 0)
   7   Spare (always 0)

geoError (2-byte integer, array size: nscan): A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector \( \mathbf{v} \) from the satellite forward direction of motion, measured clockwise facing down. We define \( \mathbf{v} \) in the same direction as the spacecraft axis \( +X \), which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>
**targetSelectionMidScan** (1-byte integer, array size: nscan):

`targetSelectionMidScan` is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**operationalMode** (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry.
`operationalMode` is used in `modeStatus`. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

**limitErrorFlag** (1-byte integer, array size: nscan):
Bit flags for every ray with information about echo power limit checks. 
limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9  Missing value

**navigation (Group in HS)**

**scPos** (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9  Missing value

**scVel** (4-byte float, array size: XYZ x nscan):
The velocity vector \(ms^{-1}\) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9  Missing value

**scLat** (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9  Missing value

**scLon** (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9  Missing value

**dprAlt** (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000
to 500000 m. Special values are defined as:
-9999.9  Missing value

**scAttRollGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttPitchGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttYawGeoc** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9  Missing value

**scAttRollGeod** (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodetic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values
range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**PRE** (Group in HS)

**elevation** (4-byte float, array size: nrayHS x nscan):
Elevation of the measurement point. It is a copy of DEMHmean of level 1B product. Values are in m. Special values are defined as:
-9999.9 Missing value

**landSurfaceType** (4-byte integer, array size: nrayHS x nscan):

Land surface type.

0 - 99 Ocean
100 - 199 Land
200 - 299 Coast
300 - 399 Inland water
-9999 Missing value

**localZenithAngle** (4-byte float, array size: nrayHS x nscan):
Local zenith angle of each ray. It is a copy of scLocalZenith of level 1B product. Values are in degree. Special values are defined as:
-9999.9 Missing value

**flagPrecip** (4-byte integer, array size: nrayHS x nscan):
Precipitation or no precipitation.

For L2 Ku and L2 Ka

0  No precipitation
1  Precipitation
-9999 Missing value

For L2 DPR

0  No precipitation by both Ku and Ka
1  Precipitation by Ka, no rain by Ku
10 Precipitation by Ku, no rain by Ka
11 Precipitation by both Ku and Ka
-9999 Missing value

flagSigmaZeroSaturation (1-byte char, array size: nrayHS x nscan):

A flag to show whether echoPower is under a saturated level or not at a range bin with a calculation of sigmaZeroMeasured. Values are:

0 : normal (under saturated level)
1 : possible saturated level at real surface
2 : saturated level at real surface
99 : missing

binRealSurface (2-byte integer, array size: nrayHS x nscan):
Range bin number for real surface. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

binStormTop (2-byte integer, array size: nrayHS x nscan):
Range bin number for the storm top. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999 Missing value

heightStormTop (4-byte float, array size: nrayHS x nscan):
Height of storm top. Values are in m. Special values are defined as:
-9999.9 Missing value
**height** (4-byte float, array size: nbinHS x nrayHS x nscan):
Height. Values are in m. Special values are defined as:
-9999.9 Missing value

**binClutterFreeBottom** (2-byte integer, array size: nrayHS x nscan):
Range bin number for clutter free bottom. Special values are defined as:
-9999 Missing value

**sigmaZeroMeasured** (4-byte float, array size: nrayHS x nscan):
Surface backscattering cross section without attenuation correction (as measured). Values are in dB. Special values are defined as:
-9999.9 Missing value

**zFactorMeasured** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of reflectivity factor without attenuation correction (as measured). Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**ellipsoidBinOffset** (4-byte float, array size: nrayHS x nscan):
Distance between the ellipsoid and a center range bin of binEllipsoid defined by level 1B algorithm.

\[
\text{ellipsoidBinOffset} = \text{scRangeEllipsoid} - \{ \text{startBinRange} + (\text{binEllipsoid}-1) \times \text{rangeBinSize} \}
\]

-9999 Missing value

**snRatioAtRealSurface** (4-byte float, array size: nrayHS x nscan):
Signal/Noise ratio at real surface range bin.

\[
\text{snRatioAtRealSurface} = 10. \times \log_{10}(\text{echoPowertrueV[mW]} / \text{noisePowertrueV[mW]})
\]

-9999 Missing value

**adjustFactor** (4-byte float, array size: nrayHS x nscan):
Adjustment factor (dB) for zFactorMeasured (dBZm') and sigmaZeroMeasured (dBs0m'). dBZm' and dBs0m' are used and stored as follows:

\[
dBZm' = dBZm - \text{adjustFactor}
\]
\[
\text{dBs}_0 \text{m}' = \text{dBs}_0 \text{m} - \text{adjustFactor}
\]
The adjustment factor is the sum of 3 components:
base adjustment for instrument dependency,
angle-bin adjustment for angle-bin dependency, and
temporal adjustment for orbit number dependency.

\textbf{snowIceCover} (1-byte integer, array size: nrayHS x nscan):
TBD. Special values are defined as:
-99 Missing value

\textbf{VER} (Group in HS)

\textbf{airTemperature} (4-byte float, array size: nbinHS x nrayHS x nscan):
Air Temperature. Values are in K. Special values are defined as:
-9999.9 Missing value

\textbf{binZeroDeg} (2-byte integer, array size: nrayHS x nscan):

Range bin number with 0 degrees C level.
For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
Special values are:
177: temperature at a surface is below 0 deg. C in Ku, KaMS, DPR(NS, MS).
89: temperature at a surface is below 0 deg. C in KaHS, DPR(HS).

\textbf{attenuationNP} (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of attenuation by non-precipitation particles (cloud liquid water, cloud
icewater, water vapor, and oxygen molecules). Values are in dB/km. Special values are
defined as:
-9999.9 Missing value

\textbf{piaNP} (4-byte float, array size: nNP x nrayHS x nscan):
Path integrated attenuation caused by non-precipitation particles (cloud liquid water,
cloud ice water, water vapor, and oxygen molecules). Values are in dB. Special values are
defined as:
-9999.9 Missing value
**sigmaZeroNPCorrected** (4-byte float, array size: nrayHS x nscan):
Surface backscattering cross section with attenuation correction only for non-precipitation particles. Values are in dB. Special values are defined as:
-9999.9 Missing value

**heightZeroDeg** (4-byte float, array size: nrayHS x nscan):
Height of freezing level (0 degrees C level) Values are in m. Special values are defined as:
-9999.9 Missing value

**CSF (Group in HS)**

**flagBB** (4-byte integer, array size: nrayHS x nscan):
Bright band (BB) exists or not. The definition is different for L2 DPR on the one hand and L2 Ku and L2 Ka on the other.

**L2 DPR:**
0 no Bright Band
1 Bright Band detected by Ku and DFRm
2 Bright Band detected by Ku only
3 Bright Band detected by DFRm only
-1111 No rain value
-9999 Missing value

**L2 Ku and L2 Ka:**
0 BB not detected
1 BB detected
-1111 No rain value
-9999 Missing value

**binBBPeak** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the peak of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**binBBTop** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the top of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value
**binDFRmMLBottom** (2-byte integer, array size: nrayHS x nscan):
Range bin number for melting layer bottom detected by the DFRm method.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;0</td>
<td>Range bin number when ML bottom is detected</td>
</tr>
<tr>
<td>0</td>
<td>ML bottom not detected</td>
</tr>
<tr>
<td>-1111</td>
<td>Value for no rain in MS(HS) mode at Ka band</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**binDFRmMLTop** (2-byte integer, array size: nrayHS x nscan):
Range bin number for melting layer top detected by the DFRm method.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;0</td>
<td>Range bin number when ML top is detected</td>
</tr>
<tr>
<td>0</td>
<td>ML top not detected</td>
</tr>
<tr>
<td>-1111</td>
<td>Value for no rain in MS(HS) mode at Ka band</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**binBBBottom** (2-byte integer, array size: nrayHS x nscan):
Range bin number for the bottom of bright band. For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. A value of -1111 denotes no precipitation is present. Special values are defined as:
-9999 Missing value

**flagMLquality** (1-byte char, array size: nrayHS x nscan):
ML quality flag. Special values are defined as:
255 Missing value

**heightBB** (4-byte float, array size: nrayHS x nscan):
Height of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

**widthBB** (4-byte float, array size: nrayHS x nscan):
The width of bright band. A value of -1111.1 denotes no precipitation. Values are in m. Special values are defined as:
-9999.9 Missing value

**qualityBB** (4-byte integer, array size: nrayHS x nscan):
Quality of the bright band.
When the bright band is detected, a larger positive number indicates lower confidence in the detection.
The Ku detection is clear, but the Ka and DPR detection is somewhat doubtful.

The meaning of qualityBB has not been finalized.

3  Smeared bright band
2  Not so clear bright band
1  Clear bright band
0  BB not detected in the case of rain
-1111  No rain value
-9999  Missing value

typePrecip (4-byte integer, array size: nrayHS x nscan):
Precipitation type is expressed by an 8-digit number. The three major rain categories, stratiform, onvective, and other, can be obtained as follows:

When typePrecip is greater than zero,
Major rain type = typePrecip/10000000
   = 1    stratiform
   = 2    convective
   = 3    other

-1111  No rain value
-9999  Missing value

Let abcdefgh be the 8 digit number,

abcdefgh

then
   a: Main rain type. (a=1,2,3),
   b: 0,
   c: 0,
   d: V rain type,
   e: H rain type,
   f: BB,
   g: Shallow rain,
   h: Small size cell.

The following numbers appear as Ku and Ka (MS/HS) rain types:
   ---- stratiform
where H is the rain type by H-method, and b depends on BB, x on shallow rain and y on small size cell:

- **H = 1**: stratiform by H-method,
- **H = 2**: convective by H-method,
- **H = 3**: other by H-method.

- **b = 0**: BB not detected,
- **b = 1**: BB detected.

- **x = 0**: No shallow rain,
- **x = 1**: Shallow isolated,
- **x = 3**: Shallow non-isolated.

- **y = 0**: No small size cell,
- **y = 1**: Single cell,
- **y = 2**: Small size cell consisting of two adjacent pixels.

In the DPR product, rain type by the DFRm (measured dual frequency ratio) method is also included in typePrecip and can be obtained as follows:

- **DFRm rain type** = (typePrecip%10000000)/1000000 in C
- **DFRm rain type** = (MOD(typePrecip,10000000)/1000000 in FORTRAN

- **DFRm rain type**
  - = 1 stratiform
  - = 2 convective
  - = 4 transition
  - = 8 DFRm method cannot be applicable at Part B (in this case the conventional method determines the major rain type)
  - = 9 DFRm method cannot be applicable at Part A (in this case the conventional method determines the major rain type)

- **-1111**: No rain value
- **-9999**: Missing value
If dual frequency data is not available but Ku-only or Ka-only is available, rain type is expressed by the following 8 digit number:
- 10xxxxxx --- stratiform,
- 20xxxxxx --- convective,
- 30xxxxxx --- other,

which is a copy of Ku-only module or Ka-only module.

If dual frequency data is available, rain type is expressed by:
- 1qxxxxxx --- stratiform,
- 2qxxxxxx --- convective,
- 3qxxxxxx --- other,

where q>0.

Thus, by examining q, users can understand whether data is processed by dual frequency algorithm or single frequency algorithm.

For MS and HS, DFRm method is used.

DFRm decision classifies rain type into:
- stratiform,
- convective,

and
- transition.

The DPR numbering rule can be summarized as follows:

Let opqrstuv be the 8 digit number, then
- o: Main rain type. (o=1,2,3),
- p: DFRm rain type. (p=0,1,2,4,8,9, with p=0 for single frequency data only),
- q: DFRm BB. (q=0,1),
- r: V rain type (by conventional V-method).
  Basically r=0 for inner swath and r>0 for outer swath.
  However, r>0 when only single frequency data is available,
- s: H rain type,
- t: = 0 for inner swath,
  1 when BB is detected in the outer swath.
- u: Shallow rain,
- v: Small size cell.

DFRm type can be obtained by examining p
The meaning of p is as follows:
p = 0: single frequency data only (dual frequency data not available),
  1: stratiform by DFRm method,
  2: convective by DFRm method,
  4: transition by DFRm method,
  8: DFRm decision not available,
  9: DFRm decision not available.
Note that p>0 always in DPR processing, which is different from Ku-only or Ka-only result.
In Ku-only or Ka-only rain type numbering, p=0 always.

The following numbers appear as DPR rain types:

* For NS outer swath *

--- stratiform
1901H100
19031000

--- convective
2901H1xy (x>0 or y>0, see R\_type\_classification\_dpr2)
2902Hwxy
290310xy (x>0, y>0, see R\_type\_classification\_dpr2)
290320xy

--- other
390330xy

* For NS inner swath and MS *

--- stratiform
11B0H0xy
14B01000
19001000 --- H decision only
19011000 --- MS rain >0 but no NS rain; MS V and H determine rain type
  or NS rain >0 but no MS rain; NS V and H determine rain type
19013000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
  or NS rain >0 but no MS rain; NS V and H determine rain type
19031000 --- MS rain >0 but no NS rain; MS V and H determine rain type.
  or NS rain >0 but no MS rain; NS V and H determine rain type

--- convective
2100H0xy (x>0 or y>0)
2110H00y (y>0)
2200H0xy
2210H00y
2400H0xy
2410H00y
290010xy --- H decision only (x>0 or y>0)
290020xy --- H decision only
2901H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
(x>0 or y>0 for H=1,3)
2902H0xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290310xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
290320xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type
--- other
340030xy
390030xy --- H decision only
390330xy --- MS rain >0 but no NS rain; MS V and H determine rain type
or NS rain >0 but no MS rain; NS V and H determine rain type

***************
* For HS *
***************
--- stratiform
11B0H000
14B01000
19001000 --- H decision only
--- convective
21B0H0x0 (x>0)
22B0H0x0
240010x0 (x>0, 24B010x0 with B=0)
240020x0
241010x0 (x>0, 24B010x0 with B=1)
290010x0 (x>0) --- H decision only
290020x0 --- H decision only
--- other
340030x0
390030x0 --- H decision only

where w depends on BB by conventional V-method, B on BB
by DFRm method, H on H-method, x on shallow rain
and y on small size cell:
  w = 0: BB not detected by conventional V-method,
       1: BB detected by conventional V-method.

  B = 0: BB not detected by DFRm method,
       1: BB detected by DFRm method.

  H = 1: stratiform by H-method,
       2: convective by H-method,
       3: other by H-method.

  x = 0: No shallow rain,
       1: Shallow isolated,
       3: Shallow non-isolated.

  y = 0: No small size cell,
       1: Single cell,
       2: Small size cell consisting of two adjacent pixels.
In the above, x>0 and y>0 are taken care of in the function
R\_type\_classification\_dpr2().
=====================================================================================================

qualityTypePrecip (4-byte integer, array size: nrayHS x nscan):

Quality of the precipitation type.

1   Good
-1111 No rain value
-9999 Missing value

flagShallowRain (4-byte integer, array size: nrayHS x nscan):

Type of shallow rain
0   No shallow rain
10  Shallow isolated (maybe)
11  Shallow isolated (certain)
20  Shallow non-isolated (maybe)
21  Shallow non-isolated (certain)
-1111 No rain value
-9999 Missing value
flagHeavyIcePrecip (1-byte integer, array size: nrayHS x nscan):
This flag denotes strong or severe precipitation accompanied by solid ice hydrometeors above the -10 degree C isotherm. Special values are defined as:
-99 Missing value

SRT (Group in HS)

pathAtten (4-byte float, array size: nrayHS x nscan):
The effective 2-way path integrated attenuation. Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAalt (4-byte float, array size: method x nrayHS x nscan):
The two-way path integrated attenuation (PIA) at from the each method estimate. The path-integrated attenuation from the jth method, where

PIAalt (j=1) = PIA_Ku from forward along-track spatial at kth angle bin
PIAalt (j=2) = PIA_Ku from backward along-track spatial at kth angle bin
PIAalt (j=3) = PIA_Ku from forward hybrid at kth angle bin
PIAalt (j=4) = PIA_Ku from backward hybrid at kth angle bin
PIAalt (j=5) = PIA_Ku from temporal reference at kth angle bin
PIAalt (j=6) = PIA_Ku from light-rain temporal reference at kth angle bin

Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAdw (4-byte float, array size: nrayHS x nscan):
The 2-way attenuation.

Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAhb (4-byte float, array size: nrayHS x nscan):
The 2-way attenuation of HB.

Values are in dB. Special values are defined as:
-9999.9 Missing value

PIAhybrid (4-byte float, array size: nrayHS x nscan):
The 2-way attenuation from a weighted combination of HB and SRT.
Values are in dB. Special values are defined as:
-9999.9  Missing value

**PIAweight** (4-byte float, array size: method x nrayHS x nscan):
The weights of the individual PIA Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

\[
\text{PIAweight}_j = \frac{1}{\sigma_j^2} \cdot \left( \frac{1}{\text{Sum}_j(1/\sigma_j^2)} \right)
\]

Values are in dB. Special values are defined as:
-9999.9  Missing value

**PIAweightHY** (4-byte float, array size: two x nrayHS x nscan):
The weights of the individual PIA Ku estimates used in deriving the effective path attenuation estimate, pathAtten. The sum of the weights should equal one. Where j is method and sigma_j is the standard deviation of reference data for method j.

\[
\text{PIAweight}_j = \frac{1}{\sigma_j^2} \cdot \left( \frac{1}{\text{Sum}_j(1/\sigma_j^2)} \right)
\]

Values are in dB. Special values are defined as:
-9999.9  Missing value

**refScanID** (2-byte integer, array size: nearFar x foreBack x nrayHS x nscan):
The number of scan lines between the current scan and the beginning (or end) of the along-track reference data at each angle bin. The values are computed by the equation: Current Scan Number - Reference Scan Number. The values are positive for the Forward estimates and negative for the Backward estimates. The Fortran indices for nearFar foreBack are:

1,1 - Forward - Near reference  
2,1 - Forward - Far reference  
1,2 - Backward - Near reference  
2,2 - Backward - Far reference  

Special values are defined as:
-9999  Missing value

**reliabFactor** (4-byte float, array size: nrayHS x nscan):
Reliability Factor for the effective PIA estimate, pathAtten. Special values are defined as:
-9999.9  Missing value

**reliabFactorAlt** (4-byte float, array size: method x nrayHS x nscan):
The reliability factors associated with the individual PIA estimates corresponding to PIAalt. Special values are defined as:
-9999.9  Missing value

**reliabFactorHY** (4-byte float, array size: nrayHS x nscan):
TBD.
Special values are defined as:
-9999.9 Missing value

**reliabFlag** (2-byte integer, array size: nrayHS x nscan):
The reliability flag for the effective PIA estimate (pathAtten) based on the reliability factor (Rel_eff) in reliabFactor. Reliability Flag is:

- 1 if Rel_eff > 3; PIAeff estimate is considered reliable
- 2 if 3 ≥ Rel_eff > 1; PIAeff estimate is considered marginally reliable
- 3 if Rel_eff ≤ 1; PIAeff is unreliable
- 4 if SNR_at surface < 2dB; provides a lower bound to the path-attenuation
- 9 (no-rain case)

Special values are defined as:
-9999 Missing value

**reliabFlagHY** (2-byte integer, array size: nrayHS x nscan):
TBD.

Special values are defined as:
-9999 Missing value

**stddevEff** (4-byte float, array size: nsdew x nrayHS x nscan):
The effective standard deviation of PIA-SRT computed 3 ways.

Special values are defined as:
-9999.9 Missing value

**stddevHY** (4-byte float, array size: nrayHS x nscan):
TBD.

Special values are defined as:
-9999.9 Missing value

**zeta** (4-byte float, array size: nrayHS x nscan):
The term in the HB estimate of path attenuation.
Special values are defined as:
- 9999.9  Missing value

**DSD (Group in HS)**

**phase** (1-byte char, array size: nbinHS x nrayHS x nscan):
Phase state of the precipitation. As an unsigned byte value this represents:

- \( \text{phase} < 100 \) Temperature(C) = \( \text{phase} - 100 \)
- \( \text{phase} > 200 \) Temperature(C) = \( \text{phase} - 200 \)
- \( \text{phase} = 100 \) Top of the bright band
- \( \text{phase} = 200 \) Bottom of the bright band
- \( \text{phase} = 125 \) is used for the range bins between
  the top and peak of bright band
- \( \text{phase} = 175 \) is used for the range bins between
  the peak and bottom of bright band

Integer values of phase/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

**binNode** (2-byte integer, array size: nNode x nrayHS x nscan):

The bin number of the 5 nodes defined as:

0 - Bin number of storm top.
1 - Stratiform: 500m above center of bright band.
   Convective: 750m above 0deg C level.
2 - Stratiform: center of bright band.
   Convective: 0deg C level.
3 - Stratiform: 500m below center of bright band.
   Convective: 750m below 0deg C level.
4 - Bin number of real surface equal to
   binRealSurface in PRE group.

For NS and MS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 176 at the Ellipsoid.
For HS swaths,
bin numbers are 1-based ranging
from 1 at the top of the data window
with 88 at the Ellipsoid.
-9999 - Missing

**Experimental** (Group in HS)

**precipRateESurface2** (4-byte float, array size: nrayHS x nscan):
Estimates Surface Precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface2Status** (1-byte char, array size: nrayHS x nscan):
Status of the estimated surface precipitation using alternate method. For information on this experimental field contact the Joint DPR Team. Special values are defined as:
255 Missing value

**sigmaZeroProfile** (4-byte float, array size: nbinSZPHS x nrayHS x nscan):
Surface backscattering cross section profile around the current ifov. For information on this experimental field contact the Joint DPR Team. Values are in dB. Special values are defined as:
-9999.9 Missing value

**seaIceConcentration** (4-byte float, array size: nrayHS x nscan):
Sea ice concentration estimated by Ku. For information on this experimental field contact the Joint DPR Team. Values range from 30 to 100 percent. Special values are defined as:
-9999.9 Missing value

**SLV** (Group in HS)

**flagSLV** (1-byte integer, array size: nbinHS x nrayHS x nscan):
Special values are defined as:
-99 Missing value

**paramDSD** (4-byte float, array size: nDSD x nbinHS x nrayHS x nscan):
Parameters of the drop size distribution. The first index is dBNw; the second index is Dm in mm. Special values are defined as:
-9999.9 Missing value

**binEchoBottom** (2-byte integer, array size: nrayHS x nscan):
For NS and MS swaths, bin numbers are 1-based ranging from 1 at the top of the data
window with 176 at the Ellipsoid. For HS swaths, bin numbers are 1-based ranging from 1 at the top of the data window with 88 at the Ellipsoid. Special values are defined as:
-9999  Missing value

**piaFinal** (4-byte float, array size: nrayHS x nscan):
The final estimates of path integrated attenuation caused by precipitation particles. Values are in dB. Special values are defined as:
-9999.9  Missing value

**sigmaZeroCorrected** (4-byte float, array size: nrayHS x nscan):
Surface backscatter cross section with attenuation correction. Values are in dB. Special values are defined as:
-9999.9  Missing value

**zFactorCorrected** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of reflectivity factor with attenuation correction. Values are in dBZ. Special values are defined as:
-9999.9  Missing value

**zFactorCorrectedESurface** (4-byte float, array size: nrayHS x nscan):
Reflectivity factor with attenuation correction at estimated surface. Values are in dBZ. Special values are defined as:
-9999.9  Missing value

**zFactorCorrectedNearSurface** (4-byte float, array size: nrayHS x nscan):
Reflectivity factor with attenuation correction at near surface. Values are in dBZ. Special values are defined as:
-9999.9  Missing value

**paramNUBF** (4-byte float, array size: nNUBF x nrayHS x nscan):
TBD. Special values are defined as:
-9999.9  Missing value

**precipRate** (4-byte float, array size: nbinHS x nrayHS x nscan):
Precipitation rate. Values are in mm/hr. Special values are defined as:
-9999.9  Missing value

**precipWaterIntegrated** (4-byte float, array size: LS x nrayHS x nscan):
Precipitation water vertically integrated. Values are in g/m$^2$. Special values are defined as:
-9999.9  Missing value

**qualitySLV** (4-byte integer, array size: nrayHS x nscan):
A flag to show methods in which precipRateNearSurface is retrieved. Special values are defined as:
-9999  Missing value

**precipRateNearSurface** (4-byte float, array size: nrayHS x nscan):
Precipitation rate for the near surface. Values are in mm/hr. Special values are defined
Precipitation rate for the estimated surface. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateESurface** (4-byte float, array size: nrayHS x nscan):
Average of precipitation rate for 2 to 4km height. Values are in mm/hr. Special values are defined as:
-9999.9 Missing value

**precipRateAve24** (4-byte float, array size: nrayHS x nscan):
Phase state of the precipitation at the Near-surface level. This is a copy of the phase in the DSD group at the Near-surface level. As an unsigned byte value this represents:

phaseNearSurface < 100 Temperature(C)=phaseNearSurface-100
phaseNearSurface > 200 Temperature(C)=phaseNearSurface-200
phaseNearSurface = 100 Top of the bright band
phaseNearSurface = 200 Bottom of the bright band
phaseNearSurface = 125 is used for the range bins between the top and peak of bright band
phaseNearSurface = 175 is used for the range bins between the peak and bottom of bright band

Integer values of phaseNearSurface/100 =

0 - solid
1 - mixed phase
2 - liquid
255 - Missing

**epsilon** (4-byte float, array size: nbinHS x nrayHS x nscan):
Epsilon is the indication of the adjustment away from the initial drop size distribution, epsilon = 1 is no adjustment. Special values are defined as:
-9999.9 Missing value

**FLG** (Group in HS)

**flagEcho** (1-byte integer, array size: nbinHS x nrayHS x nscan):
Flag of precipitation and main/side lobe clutter information of each range bin.

Bit Meaning
0 For L2 Ku: Precipitation judged by L2 Ku algorithm (copy of bit 2)
0 For L2 Ka: Precipitation judged by L2 Ka algorithm (copy of bit 3)
0 For L2 DPR: Precipitation judged by L2 DPR algorithm (copy of bit 1)
1 Precipitation judged by L2 DPR algorithm
2 Precipitation judged by L2 Ku algorithm
3 Precipitation judged by L2 Ka algorithm
4 Main lobe clutter judged by L2 Ku algorithm
5 Main lobe clutter judged by L2 Ka algorithm
6 Side lobe clutter judged by L2 Ku algorithm
7 Side lobe clutter judged by L2 Ka algorithm

qualityData (4-byte integer, array size: nrayHS x nscan):
Normal data gives ”0”. Non-zero values mean the kinds of errors. Special values are defined as:
-9999 Missing value

Flag of quality data. Bit range from 8 to 23 contains flags by each module. Each module flag has 2 bits of information.

The 2 bit flag for each module has values:
[higher bit lower bit]
[0 0] Good
[0 1] Warning but usable
[1 0] NG or error

The bits of qualityData are assigned as follows:

Bit Meaning
0 - 7 Copy of dataQuality in level 1B product
8 - 9 Flag by input module
10 - 11 Flag by preparation module
12 - 13 Flag by vertical module
14 - 15 Flag by classification module
16 - 17 Flag by SRT module
18 - 19 Flag by DSD module
20 - 21 Flag by solver module
22 - 23 Flag by output module
24 - 31 Spare

qualityFlag (1-byte integer, array size: nrayHS x nscan):
Flag derived from qualityData with the following values: Special values are defined as:
-99  Missing value

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>High quality. No issues.</td>
</tr>
<tr>
<td>1</td>
<td>Low quality (DPR modules had warnings but still made a retrieval)</td>
</tr>
<tr>
<td>2</td>
<td>Bad (DPR modules had errors or dataQuality is bad and retrieval is missing)</td>
</tr>
</tbody>
</table>

`flagSensor` (1-byte integer, array size: nscan):

Flag of input Ku/Ka data condition.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valid</td>
</tr>
<tr>
<td>-99</td>
<td>Invalid (judged by dataQuality)</td>
</tr>
</tbody>
</table>

`flagScanPattern` (2-byte integer, array size: nscan):

Flag of scan pattern.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TBD</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

C Structure Header file:

```c
#ifndef _TK_2ADPRX_H_
#define _TK_2ADPRX_H_

#ifndef _L2ADPRX_HS_FLG_
#define _L2ADPRX_HS_FLG_

typedef struct {
    signed char flagEcho[24][88];
    int qualityData[24];
    signed char qualityFlag[24];
    signed char flagSensor;
    short flagScanPattern;
} L2ADPRX_HS_FLG;
```
#endif

#ifndef _L2ADPRX_HS_SLV_
#define _L2ADPRX_HS_SLV_

typedef struct {
    signed char flagSLV[24][88];
    float paramDSD[24][88][2];
    short binEchoBottom[24];
    float piaFinal[24];
    float sigmaZeroCorrected[24];
    float zFactorCorrected[24][88];
    float zFactorCorrectedESurface[24];
    float zFactorCorrectedNearSurface[24];
    float paramNUBF[24][3];
    float precipRate[24][88];
    float precipWaterIntegrated[24][2];
    int qualitySLV[24];
    float precipRateNearSurface[24];
    float precipRateESurface[24];
    float precipRateAve24[24];
    unsigned char phaseNearSurface[24];
    float epsilon[24][88];
} L2ADPRX_HS_SLV;

#endif

#ifndef _L2ADPRX_HS_EXPERIMENTAL_
#define _L2ADPRX_HS_EXPERIMENTAL_

typedef struct {
    float precipRateESurface2[24];
    unsigned char precipRateESurface2Status[24];
    float sigmaZeroProfile[24][5];
    float seaIceConcentration[24];
} L2ADPRX_HS_EXPERIMENTAL;

#endif

#ifndef _L2ADPRX_HS_DSD_
#define _L2ADPRX_HS_DSD_

typedef struct {

unsigned char phase[24][88];
short binNode[24][5];
} L2ADPRX_HS_DSD;
#endif

#ifndef _L2ADPRX_HS_SRT_
define _L2ADPRX_HS_SRT_

typedef struct {
    float pathAtten[24];
    float PIAalt[24][6];
    float PIAdw[24];
    float PIAhb[24];
    float PIAhybrid[24];
    float PIAweight[24][6];
    float PIAwightHY[24][2];
    short refScanID[24][2][2];
    float reliabFactor[24];
    float reliabFactorAlt[24][6];
    float reliabFactorHY[24];
    short reliabFlag[24];
    short reliabFlagHY[24];
    float stddevEff[24][3];
    float stddevHY[24];
    float zeta[24];
} L2ADPRX_HS_SRT;
#endif

#ifndef _L2ADPRX_HS_CSF_
define _L2ADPRX_HS_CSF_

typedef struct {
    int flagBB[24];
    short binBBPeak[24];
    short binBBTop[24];
    short binDFRmMLBottom[24];
    short binDFRmMLTop[24];
    short binBBBottom[24];
    unsigned char flagMLquality[24];
    float heightBB[24];
    float widthBB[24];
typedef struct {
  float airTemperature[24][88];
  short binZeroDeg[24];
  float attenuationNP[24][88];
  float piaNP[24][4];
  float sigmaZeroNPCorrected[24];
  float heightZeroDeg[24];
} L2ADPRX_HS_VER;

typedef struct {
  float elevation[24];
  int landSurfaceType[24];
  float localZenithAngle[24];
  int flagPrecip[24];
  unsigned char flagSigmaZeroSaturation[24];
  short binRealSurface[24];
  short binStormTop[24];
  float heightStormTop[24];
  float height[24][88];
  short binClutterFreeBottom[24];
  float sigmaZeroMeasured[24];
  float zFactorMeasured[24][88];
  float ellipsoidBinOffset[24];
  float snRatioAtRealSurface[24];
  float adjustFactor[24];
  signed char snowIceCover[24];
} L2ADPRX_HS_VER;
typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2ADPRX_HS_SCANSTATUS;

#endif
#endif

typedef struct {
    SCANTIME ScanTime;
    float Latitude[24];
    float Longitude[24];
    L2ADPRX_HS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2ADPRX_HS_PRE PRE;
    L2ADPRX_HS_VER VER;
    L2ADPRX_HS_CSF CSF;
    L2ADPRX_HS_SRT SRT;
    L2ADPRX_HS_DSD DSD;
    L2ADPRX_HS_EXPERIMENTAL Experimental;
    L2ADPRX_HS_SLV SLV;
    L2ADPRX_HS_FLG FLG;
} L2ADPRX_HS;
#ifndef _L2ADPRX_FS_TRG_
#define _L2ADPRX_FS_TRG_

typedef struct {
    float NUBFindex[49];
    unsigned char MSindex[49];
    signed char MSindexKu[49];
    signed char MSindexKa[49];
    unsigned char precipFrac[49][3];
    float RNUBFcond[49];
    unsigned char MSsurfPeakIndexKu[49];
    unsigned char MSsurfPeakIndexKa[49];
    unsigned char MStroughsurfIndexKu[49];
    unsigned char MStroughsurfIndexKa[49];
    unsigned char MSkneeDFRindex[49];
    unsigned char MSThrZindex[49];
    unsigned char NUBFratioPIAindex[49];
    unsigned char NUBFnZmVarIndex[49][3];
    unsigned char NUBFnZkVarIndex[49][3];
    short NUBFnZmVarScaling[49];
    short NUBFnZkVarScaling[49];
    float NUBFsrfSliceIndex[49][30];
    float NUBFprofZPC[49][30];
    short MSbreakpoints[49][13];
    float MSslopes[49][10];
    float MSslopePoints[49][13];
    float MSslopeFits[49][6];
    unsigned char MSslowSNRrangeFilter[49][4];
    float NUBFcorrPIA[49][2];
    float triggerParameters[49][8];
} L2ADPRX_FS_TRG;

#define _L2ADPRX_FS_FLG_

typedef struct {
    signed char flagEcho[49][176];
    int qualityData[49];
signed char qualityFlag[49][2];
signed char flagSensor[2];
short flagScanPattern[2];
} L2ADPRX_FS_FLG;

#endif

#ifndef _L2ADPRX_FS_SLV_
#define _L2ADPRX_FS_SLV_

typedef struct {
    signed char flagSLV[49][176];
    float paramDSD[49][176][2];
    short binEchoBottom[49];
    float piaFinal[49][2];
    float sigmaZeroCorrected[49][2];
    float zFactorCorrected[49][176][2];
    float zFactorCorrectedESurface[49][2];
    float zFactorCorrectedNearSurface[49][2];
    float paramNUBF[49][3];
    float precipRate[49][176];
    float precipWaterIntegrated[49][2];
    int qualitySLV[49];
    float precipRateNearSurface[49];
    float precipRateESurface[49];
    float precipRateAve24[49];
    unsigned char phaseNearSurface[49];
    float epsilon[49][176];
} L2ADPRX_FS_SLV;

#endif

#ifndef _L2ADPRX_FS_EXPERIMENTAL_
#define _L2ADPRX_FS_EXPERIMENTAL_

typedef struct {
    float precipRateESurface2[49];
    unsigned char precipRateESurface2Status[49];
    float sigmaZeroProfile[49][7][2];
    float seaIceConcentration[49];
    unsigned char flagSurfaceSnowfall[49];
    unsigned char flagGraupelHail[49];
    short binMixedPhaseTop[49];
typedef struct {
    float pathAtten[49][2];
    float PIAalt[49][6][2];
    float PIAdv[49][2];
    float PIAhb[49][2];
    float PIAhb[49][2];
    float PIAhybrid[49][2];
    float PIAweight[49][6];
    float PIAweightHY[49][3];
    short refScanID[49][2][2];
    float reliabFactor[49];
    float reliabFactorAlt[49][6];
    float reliabFactorHY[49];
    short reliabFlag[49];
    short reliabFlagHY[49];
    float stddevEff[49][3][2];
    float stddevHY[49][2];
    float zeta[49][2];
} L2ADPRX_FS_SRT;
int flagBB[49];
short binBBPeak[49];
short binBBTop[49];
short binDFRmMLBottom[49];
short binDFRmMLTop[49];
short binBBBottom[49];
short binHeavyIcePrecipTop[49][3];
short binHeavyIcePrecipBottom[49][3];
unsigned char nHeavyIcePrecip[49][3];
unsigned char flagMLquality[49];
float heightBB[49];
float widthBB[49];
int qualityBB[49];
int typePrecip[49];
int qualityTypePrecip[49];
int flagShallowRain[49];
signed char flagHeavyIcePrecip[49];
signed char flagAnvil[49];
} L2ADPRX_FS_CSF;

#endif

#ifndef _L2ADPRX_FS_VER_
#define _L2ADPRX_FS_VER_

typedef struct {
    float airTemperature[49][176];
    short binZeroDeg[49];
    float attenuationNP[49][176][2];
    float piaNP[49][4][2];
    float sigmaZeroNPCorrected[49][2];
    float heightZeroDeg[49];
} L2ADPRX_FS_VER;

#endif

#ifndef _L2ADPRX_FS_PRE_
#define _L2ADPRX_FS_PRE_

typedef struct {
    float elevation[49];
    int landSurfaceType[49];
    float localZenithAngle[49][2];
} L2ADPRX_FS_PRE;

#endif
int flagPrecip[49];
unsigned char flagSigmaZeroSaturation[49][2];
short binRealSurface[49][2];
short binStormTop[49];
float heightStormTop[49];
float height[49][176];
short binClutterFreeBottom[49];
float sigmaZeroMeasured[49][2];
float zFactorMeasured[49][176][2];
float ellipsoidBinOffset[49][2];
float snRatioAtRealSurface[49][2];
float adjustFactor[49][2];
signed char snowIceCover[49];
} L2ADPRX_FS_PRE;

#endif

#ifndef _NAVIGATION_
define _NAVIGATION_

typedef struct {
  float scPos[3];
  float scVel[3];
  float scLat;
  float scLon;
  float scAlt;
  float dprAlt;
  float scAttRollGeoc;
  float scAttPitchGeoc;
  float scAttYawGeoc;
  float scAttRollGeod;
  float scAttPitchGeod;
  float scAttYawGeod;
  float greenHourAng;
  double timeMidScan;
  double timeMidScanOffset;
} NAVIGATION;
#endif

#ifndef _L2ADPRX_FS_SCANSTATUS_
define _L2ADPRX_FS_SCANSTATUS_

} L2ADPRX_FS_SCANSTATUS;
typedef struct {
    signed char dataQuality[2];
    signed char dataWarning[2];
    signed char missing[2];
    signed char modeStatus[2];
    short geoError[2];
    short geoWarning[2];
    short SCorientation;
    short pointingStatus[2];
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode[2];
    signed char limitErrorFlag[2];
    double FractionalGranuleNumber;
} L2ADPRX_FS_SCANSTATUS;

#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L2ADPRX_FS_
#define _L2ADPRX_FS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2ADPRX_FS_SCANSTATUS scanStatus;
}
NAVIGATION navigation;
L2ADPRX_FS_PRE PRE;
L2ADPRX_FS_VER VER;
L2ADPRX_FS_CSF CSF;
L2ADPRX_FS_SRT SRT;
L2ADPRX_FS_DSD DSD;
L2ADPRX_FS_EXPERIMENTAL Experimental;
L2ADPRX_FS_SLV SLV;
L2ADPRX_FS_FLG FLG;
L2ADPRX_FS_TRG TRG;
} L2ADPRX_FS;
#endif

#ifndef _L2ADPRX_SWATHS_
define _L2ADPRX_SWATHS_

typedef struct {
    L2ADPRX_FS FS;
    L2ADPRX_HS HS;
} L2ADPRX_SWATHS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L2ADPRX_HS_FLG/
    BYTE flagEcho(88,24)
    INTEGER*4 qualityData(24)
    BYTE qualityFlag(24)
    BYTE flagSensor
    INTEGER*2 flagScanPattern
END STRUCTURE

STRUCTURE /L2ADPRX_HS_SLV/
    BYTE flagSLV(88,24)
    REAL*4 paramDSD(2,88,24)
    INTEGER*2 binEchoBottom(24)
    REAL*4 piaFinal(24)
    REAL*4 sigmaZeroCorrected(24)
    REAL*4 zFactorCorrected(88,24)
REAL*4 zFactorCorrectedESurface(24)
REAL*4 zFactorCorrectedNearSurface(24)
REAL*4 paramNUBF(3,24)
REAL*4 precipRate(88,24)
REAL*4 precipWaterIntegrated(2,24)
INTEGER*4 qualitySLV(24)
REAL*4 precipRateNearSurface(24)
REAL*4 precipRateESurface(24)
REAL*4 precipRateAve24(24)
CHARACTER phaseNearSurface(24)
REAL*4 epsilon(88,24)
END STRUCTURE

STRUCTURE /L2ADPRX_HS_EXPERIMENTAL/
  REAL*4 precipRateESurface2(24)
  CHARACTER precipRateESurface2Status(24)
  REAL*4 sigmaZeroProfile(5,24)
  REAL*4 seaIceConcentration(24)
END STRUCTURE

STRUCTURE /L2ADPRX_HS_DSD/
  CHARACTER phase(88,24)
  INTEGER*2 binNode(5,24)
END STRUCTURE

STRUCTURE /L2ADPRX_HS_SRT/
  REAL*4 pathAtten(24)
  REAL*4 PIAalt(6,24)
  REAL*4 PIAdw(24)
  REAL*4 PIAhb(24)
  REAL*4 PIAhybrid(24)
  REAL*4 PIAweight(6,24)
  REAL*4 PIAweightHY(2,24)
  INTEGER*2 refScanID(2,2,24)
  REAL*4 reliabFactor(24)
  REAL*4 reliabFactorAlt(6,24)
  REAL*4 reliabFactorHY(24)
  INTEGER*2 reliabFlag(24)
  INTEGER*2 reliabFlagHY(24)
  REAL*4 stddevEff(3,24)
  REAL*4 stddevHY(24)
  REAL*4 zeta(24)
END STRUCTURE
STRUCTURE /L2ADPRX_HS_CS/F
    INTEGER*4 flagBB(24)
    INTEGER*2 binBBPeak(24)
    INTEGER*2 binBBTop(24)
    INTEGER*2 binDFRmMLBottom(24)
    INTEGER*2 binDFRmMLTop(24)
    INTEGER*2 binBBBottom(24)
    CHARACTER flagMLquality(24)
    REAL*4 heightBB(24)
    REAL*4 widthBB(24)
    INTEGER*4 qualityBB(24)
    INTEGER*4 typePrecip(24)
    INTEGER*4 qualityTypePrecip(24)
    INTEGER*4 flagShallowRain(24)
    BYTE flagHeavyIcePrecip(24)
END STRUCTURE

STRUCTURE /L2ADPRX_HS_VER/
    REAL*4 airTemperature(88,24)
    INTEGER*2 binZeroDeg(24)
    REAL*4 attenuationNP(88,24)
    REAL*4 piaNP(4,24)
    REAL*4 sigmaZeroNPCorrected(24)
    REAL*4 heightZeroDeg(24)
END STRUCTURE

STRUCTURE /L2ADPRX_HS_PRE/
    REAL*4 elevation(24)
    INTEGER*4 landSurfaceType(24)
    REAL*4 localZenithAngle(24)
    INTEGER*4 flagPrecip(24)
    CHARACTER flagSigmaZeroSaturation(24)
    INTEGER*2 binRealSurface(24)
    INTEGER*2 binStormTop(24)
    REAL*4 heightStormTop(24)
    REAL*4 height(88,24)
    INTEGER*2 binClutterFreeBottom(24)
    REAL*4 sigmaZeroMeasured(24)
    REAL*4 zFactorMeasured(88,24)
    REAL*4 ellipsoidBinOffset(24)
    REAL*4 snRatioAtRealSurface(24)
    REAL*4 adjustFactor(24)
BYTE snowIceCover(24)
END STRUCTURE

STRUCTURE /L2ADPRX_HS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2ADPRX_HS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(24)
  REAL*4 Longitude(24)
  RECORD /L2ADPRX_HS_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L2ADPRX_HS_PRE/ PRE
  RECORD /L2ADPRX_HS_VER/ VER
  RECORD /L2ADPRX_HS_CSF/ CSF
  RECORD /L2ADPRX_HS_SRT/ SRT
  RECORD /L2ADPRX_HS_DSD/ DSD
  RECORD /L2ADPRX_HS_EXPERIMENTAL/ Experimental
  RECORD /L2ADPRX_HS_SLV/ SLV
  RECORD /L2ADPRX_HS_FLG/ FLG
END STRUCTURE

STRUCTURE /L2ADPRX_FS_TRG/
  REAL*4 NUBFindex(49)
  CHARACTER MSindex(49)
  BYTE MSindexKu(49)
  BYTE MSindexKa(49)
  CHARACTER precipFrac(3,49)
  REAL*4 RNUBFcond(49)
  CHARACTER MSsurfPeakIndexKu(49)
CHARACTER MSsurfPeakIndexKa(49)
CHARACTER MSTthroughsurfIndexKu(49)
CHARACTER MSTthroughsurfIndexKa(49)
CHARACTER MSkneeDFRindex(49)
CHARACTER MSThrZindex(49)
CHARACTER NUBFratioPIAindex(49)
CHARACTER NUBFnZmVarIndex(3,49)
CHARACTER NUBFnZkVarIndex(3,49)
INTEGER*2 NUBFnZmVarScaling(49)
INTEGER*2 NUBFnZkVarScaling(49)
REAL*4 NUBFsurfSliceIndex(30,49)
REAL*4 NUBFprofZPC(30,49)
INTEGER*2 MSbreakpoints(13,49)
REAL*4 MSslopes(10,49)
REAL*4 MSslopePoints(13,49)
REAL*4 MSslopeFits(6,49)
CHARACTER MSlowSNRrangeFilter(4,49)
REAL*4 NUBFcorrPIA(2,49)
REAL*4 triggerParameters(8,49)

END STRUCTURE

STRUCTURE /L2ADPRX_FS_FLG/
  BYTE flagEcho(176,49)
  INTEGER*4 qualityData(49)
  BYTE qualityFlag(2,49)
  BYTE flagSensor(2)
  INTEGER*2 flagScanPattern(2)
END STRUCTURE

STRUCTURE /L2ADPRX_FS_SLV/
  BYTE flagSLV(176,49)
  REAL*4 paramDSD(2,176,49)
  INTEGER*2 binEchoBottom(49)
  REAL*4 piaFinal(2,49)
  REAL*4 sigmaZeroCorrected(2,49)
  REAL*4 zFactorCorrected(2,176,49)
  REAL*4 zFactorCorrectedESurface(2,49)
  REAL*4 zFactorCorrectedNearSurface(2,49)
  REAL*4 paramNUBF(3,49)
  REAL*4 precipRate(176,49)
  REAL*4 precipWaterIntegrated(2,49)
  INTEGER*4 qualitySLV(49)
  REAL*4 precipRateNearSurface(49)
REAL*4 precipRateESurface(49)
REAL*4 precipRateAve24(49)
CHARACTER phaseNearSurface(49)
REAL*4 epsilon(176,49)

END STRUCTURE

STRUCTURE /L2ADPRX_FS_EXPERIMENTAL/
  REAL*4 precipRateESurface2(49)
  CHARACTER precipRateESurface2Status(49)
  REAL*4 sigmaZeroProfile(2,7,49)
  REAL*4 seaIceConcentration(49)
  CHARACTER flagSurfaceSnowfall(49)
  CHARACTER flagGraupelHail(49)
  INTEGER*2 binMixedPhaseTop(49)
  REAL*4 surfaceSnowfallIndex(49)

END STRUCTURE

STRUCTURE /L2ADPRX_FS_DSD/
  CHARACTER phase(176,49)
  INTEGER*2 binNode(5,49)

END STRUCTURE

STRUCTURE /L2ADPRX_FS_SRT/
  REAL*4 pathAtten(2,49)
  REAL*4 PIAalt(2,6,49)
  REAL*4 PIAdw(2,49)
  REAL*4 PIAhb(2,49)
  REAL*4 PIAhybrid(2,49)
  REAL*4 PIAweight(6,49)
  REAL*4 PIAweightHY(3,49)
  INTEGER*2 refScanID(2,2,49)
  REAL*4 reliabFactor(49)
  REAL*4 reliabFactorAlt(6,49)
  REAL*4 reliabFactorHY(49)
  INTEGER*2 reliabFlag(49)
  INTEGER*2 reliabFlagHY(49)
  REAL*4 stddevEff(2,3,49)
  REAL*4 stddevHY(2,49)
  REAL*4 zeta(2,49)

END STRUCTURE

STRUCTURE /L2ADPRX_FS_CSF/
  INTEGER*4 flagBB(49)
INTEGER*2 binBBPeak(49)
INTEGER*2 binBBTop(49)
INTEGER*2 binDFRmMLBottom(49)
INTEGER*2 binDFRmMLTop(49)
INTEGER*2 binBBBottom(49)
INTEGER*2 binHeavyIcePrecipTop(3,49)
INTEGER*2 binHeavyIcePrecipBottom(3,49)
CHARACTER nHeavyIcePrecip(3,49)
CHARACTER flagMLquality(49)
REAL*4 heightBB(49)
REAL*4 widthBB(49)
INTEGER*4 qualityBB(49)
INTEGER*4 typePrecip(49)
INTEGER*4 qualityTypePrecip(49)
INTEGER*4 flagShallowRain(49)
BYTE flagHeavyIcePrecip(49)
BYTE flagAnvil(49)
END STRUCTURE

STRUCTURE /L2ADPRX_FS_VER/
    REAL*4 airTemperature(176,49)
    INTEGER*2 binZeroDeg(49)
    REAL*4 attenuationNP(2,176,49)
    REAL*4 piaNP(2,4,49)
    REAL*4 sigmaZeroNPCorrected(2,49)
    REAL*4 heightZeroDeg(49)
END STRUCTURE

STRUCTURE /L2ADPRX_FS_PRE/
    REAL*4 elevation(49)
    INTEGER*4 landSurfaceType(49)
    REAL*4 localZenithAngle(2,49)
    INTEGER*4 flagPrecip(49)
    CHARACTER flagSigmaZeroSaturation(2,49)
    INTEGER*2 binRealSurface(2,49)
    INTEGER*2 binStormTop(49)
    REAL*4 heightStormTop(49)
    REAL*4 height(176,49)
    INTEGER*2 binClutterFreeBottom(49)
    REAL*4 sigmaZeroMeasured(2,49)
    REAL*4 zFactorMeasured(2,176,49)
    REAL*4 ellipsoidBinOffset(2,49)
    REAL*4 snRatioAtRealSurface(2,49)
REAL*4 adjustFactor(2,49)
BYTE snowIceCover(49)

END STRUCTURE

STRUCTURE /NAVIGATION/
REAL*4 scPos(3)
REAL*4 scVel(3)
REAL*4 scLat
REAL*4 scLon
REAL*4 scAlt
REAL*4 dprAlt
REAL*4 scAttRollGeoc
REAL*4 scAttPitchGeoc
REAL*4 scAttYawGeoc
REAL*4 scAttRollGeod
REAL*4 scAttPitchGeod
REAL*4 scAttYawGeod
REAL*4 greenHourAng
REAL*8 timeMidScan
REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L2ADPRX_FS_SCANSTATUS/
BYTE dataQuality(2)
BYTE dataWarning(2)
BYTE missing(2)
BYTE modeStatus(2)
INTEGER*2 geoError(2)
INTEGER*2 geoWarning(2)
INTEGER*2 SCorientation
INTEGER*2 pointingStatus(2)
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode(2)
BYTE limitErrorFlag(2)
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /SCANTIME/
INTEGER*2 Year
BYTE Month
BYTE Day0fMonth
BYTE Hour
BYTE Minute
BYTE Second
INTEGER*2 MilliSecond
INTEGER*2 DayOfYear
REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2ADPRX_FS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
  REAL*4 Longitude(49)
  RECORD /L2ADPRX_FS_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L2ADPRX_FS_PRE/ PRE
  RECORD /L2ADPRX_FS_VER/ VER
  RECORD /L2ADPRX_FS_CSFIN/ CSF
  RECORD /L2ADPRX_FS_SRT/ SRT
  RECORD /L2ADPRX_FS_DSD/ DSD
  RECORD /L2ADPRX_FS_EXPERIMENTAL/ Experimental
  RECORD /L2ADPRX_FS_SLV/ SLV
  RECORD /L2ADPRX_FS_FLG/ FLG
  RECORD /L2ADPRX_FS_TRG/ TRG
END STRUCTURE

STRUCTURE /L2ADPRX_SWATHS/
  RECORD /L2ADPRX_FS/ FS;
  RECORD /L2ADPRX_HS/ HS;
END STRUCTURE

5.63 2AKuTMPX - Ku Temporary

The 2AKuTMP product contains intermediate data used in the 2AKu retrieval.

Dimension definitions:
nscan  var  Number of scans in the granule.
nray   49  Number of angle bins in each NS scan.
nrayMS 25  Number of angle bins in each MS scan.
nrayHS 24  Number of angle bins in each HS scan.
nbin   176 Number of range bins in each NS and MS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
nbinln 260 Number of L1 range bins in each NS and MS ray. Bin interval is 125 m.
nbinHS  88 Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.
nNP    4  Number of NP kinds.
nRScan 4  Number of Ref Scan ID.
method 6  Number of SRT methods.
nNode  5  Number of binNode.
nDSD   2  Number of DSD parameters. Parameters are N0 and D0.
LS     2  Liquid, solid.
nDielec 2  Number of dielectric constants.
nParmFV 2  Number of parameters of falling velocity.
piaNPGd 4  Number of parameters of piaNPGANAL.

Figure 919 through Figure 928 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo (Metadata):**
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.
Figure 919: Data Format Structure for 2AKuTMPX, Ku Temporary

Figure 920: Data Format Structure for 2AKuTMPX, ScanTime
**FS (Swath)**

**SwathHeader (Metadata):**
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime (Group)**
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value
Figure 922: Data Format Structure for 2AKuTMPX, VertLocate

Figure 923: Data Format Structure for 2AKuTMPX, Transmitter
Figure 924: Data Format Structure for 2AKuTMPX, Receiver

Figure 925: Data Format Structure for 2AKuTMPX, PRETMP

Figure 926: Data Format Structure for 2AKuTMPX, VERTMP
Figure 927: Data Format Structure for 2AKuTMPX, DSDTMP

- particleTemperature: 4 bytes, Array: nbin x nray x nscan
- flagDSD: 4 bytes, Array: nbin x nray x nscan
- attenParmAlpha: 4 bytes, Array: nbin x nray x nscan
- attenParmBeta: 4 bytes, Array: nbin x nray x nscan
- reliabEpsilon: 4 bytes, Array: nray x nscan

Figure 928: Data Format Structure for 2AKuTMPX, SLVTMP

- attenuationPrecip: 4 bytes, Array: nbin x nray x nscan
- precipWater: 4 bytes, Array: nbin x nray x nscan
- zFactorForward1: 4 bytes, Array: nbin x nray x nscan
- zFactorForward2: 4 bytes, Array: nbin x nray x nscan

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousands of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value
**Longitude** (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
13  Spare (always 0)
14  Spare (always 0)
15  Spare (always 0)

**geoWarning** (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9   Missing value

VertLocate (Group)
landOceanFlag (2-byte integer, array size: nray x nscan):
Land or ocean information. The values of the flag are:

- 0 = Water
- 1 = Land
- 2 = Coast
- 3 = Water (w/ large attenuation)
- 4 = Land/Coast (w/ large attenuation)

Values range from 0 to 6. Special values are defined as:
-9999 Missing value

scLocalZenith (4-byte float, array size: nray x nscan):
The angle, in degrees, between the local zenith and the beam’s center line. The local
(geodetic) zenith at the intersection of the ray and the earth ellipsoid is used. Values
range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

startBinRange (4-byte float, array size: nray x nscan):
The range to the first radar bin. Values range from 350000 to 500000 m. Special values
are defined as:
-9999.9 Missing value

echoHighResBinNumber (2-byte integer, array size: nray x nscan):
The bin number of the first radar bin with detectable echo. Values range from 0 to 260
range bin number. Special values are defined as:
-9999 Missing value

echoLowResBinNumber (2-byte integer, array size: nray x nscan):
The bin number of the first radar bin with detectable echo at low threshold. Values range
from 0 to 260 range bin number. Special values are defined as:
-9999 Missing value

binEllipsoid (2-byte integer, array size: nray x nscan):
The range bin number of the earth ellipsoid. Values range from 1 to 260 range bin number.
Special values are defined as:
-9999 Missing value

scRangeEllipsoid (4-byte float, array size: nray x nscan):
The spacecraft range to the Ellipsoid. Values range from 0 to 500000 m. Special values
are defined as:
-9999.9 Missing value

binDEM (2-byte integer, array size: nray x nscan):
The bin number of the surface from digital elevation model data. Values range from 1 to
260 range bin number. Special values are defined as:
-9999 Missing value
**scRangeDEM** (4-byte float, array size: nray x nscan):
The range to the surface from digital elevation model data. Values range from 0 to 500000 m. Special values are defined as:
- 9999.9 Missing value

**DEMHarange** (2-byte integer, array size: nray x nscan):
The mean of the height above Ellipsoid from digital elevation model data within the radar footprint. Values range from 0 to 9000 m. Special values are defined as:
- 9999 Missing value

**binDEMHtop** (2-byte integer, array size: nray x nscan):
The range bin number of the maximum DEM surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11km. Values range from 1 to 260 range bin number. Special values are defined as:
- 9999 Missing value

**binDEMHbottom** (2-byte integer, array size: nray x nscan):
The range bin number of the minimum DEM surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11km. Values range from 1 to 260 range bin number. Special values are defined as:
- 9999 Missing value

**binEchoPeak** (2-byte integer, array size: nray x nscan):
The bin number of the peak echo in along the slant range radar profile. Values range from 1 to 260 range bin number. Special values are defined as:
- 9999 Missing value

**alongTrackBeamWidth** (4-byte float, array size: nray x nscan):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power in the along-track direction.

**crossTrackBeamWidth** (4-byte float, array size: nray x nscan):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power along the cross-track direction.

**mainlobeEdge** (2-byte integer, array size: nray x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

**sidelobeRange** (2-byte integer, array size: nray x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter position from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

**ellipsoidBinOffset** (4-byte float, array size: nray x nscan):
The offset of the Ellipsoid bin from the actual surface.

**rangeBinSize** (4-byte float, array size: nscan):
The size of the radar range bin (gate) along the direction of propagation.
ratioLand (1-byte integer, array size: nray x nscan):
The ratio of land to all other surface types within the radar footprint.

ratioOcean (1-byte integer, array size: nray x nscan):
The ratio of ocean to all other surface types within the radar footprint.

ratioInLand (1-byte integer, array size: nray x nscan):
The ratio of inland water to all other surface types within the radar footprint.

ratioCoast (1-byte integer, array size: nray x nscan):
The ratio of coast to all other surface types within the radar footprint.

Transmitter (Group)

radarTransPower (4-byte float, array size: nscan):
The total (sum) power of 128 SSPA elements corrected with SSPA temperature in orbit, based on temperature test data of SSPA transmission power. Values are in dBm. Special values are defined as:
-9999.9 Missing value

transPulseWidth (4-byte float, array size: nscan):
Transmitted pulse width corrected with FCIF temperature in orbit, based on temperature test data of FCIF. Values range from 0.0000015 to 0.0000017 s. Special values are defined as:
-9999.9 Missing value

taxAntGain (4-byte float, array size: nray x nscan):
Transmitted radar antenna effectiveness (dB).

Receiver (Group)

echoCount (1-byte char, array size: nbinln x nray x nscan):
Special values are defined as:
255 Missing value

noiseCount (4-byte float, array size: nray x nscan):
Special values are defined as:
-9999.9 Missing value

echoPower (2-byte integer, array size: nbinln x nray x nscan):
Return power. Values in dBm are multiplied by 100 and stored in the file as a 2-byte integer. The unit in the file is thus 0.01 dBm. Bins where data is not written due to a transmission, calibration, or other problem, including an entire scan of missing bins, have the value of -32734. The range is -120 dBm to -20 dBm, which corresponds to values in the file from -12000 to -2000.
**noisePower** (2-byte integer, array size: nray x nscan):
An average of the 4 measured system noise values. Values in dBm are multiplied by 100
and stored in the file as a 2-byte integer. The unit in the file is thus 0.01 dBm. The range
is -120 dBm to -20 dBm which corresponds to values in the file from -12000 to -2000. The
accuracy is 0.9 dBm. Missing data are given the value of -32,734.

**noiseSampleNumber** (2-byte integer, array size: nray x nscan):
The number of noise samples used. Values range from 0 to 140 Number. Special values
are defined as:
-9999  Missing value

**echoSampleNumber** (1-byte integer, array size: nray x nscan):
Sample number of echo power stored in VPRF table. Values range from 0 to 127 Number.
Special values are defined as:
-99  Missing value

**rxAntGain** (4-byte float, array size: nray x nscan):
Received radar antenna effectiveness (dB).

**receivedPulseWidth** (4-byte float, array size: nscan):
Received pulse width (dBm).

---

**PRETMP** (Group)

**sidelobePower** (4-byte float, array size: nbin x nray x nscan):
A power of sidelobe clutter estimated in the algorithm. Values are in dBm. Special values
are defined as:
-9999.9  Missing value

**sidelobePowerError** (4-byte float, array size: nbin x nray x nscan):
An error of a power of sidelobe clutter estimated in the algorithm. Values are in dBm.
Special values are defined as:
-9999.9  Missing value

**flagSidelobePower** (2-byte integer, array size: nray x nscan):
A flag to show a selected table of a sidelobe database. Special values are defined as:
-9999  Missing value

**binRangeBottom** (4-byte integer, array size: nray x nscan):
Range bin number (1-260 for Ku, KaMS or 1-130 for KaHS) of level 1 corresponding to
the bottom of level 2. Special values are defined as:
-9999  Missing value

**binRangeTop** (4-byte integer, array size: nray x nscan):
Range bin number (1-260 for Ku, KaMS or 1-130 for KaHS) of level 1 corresponding to
the top of level 2. Special values are defined as:
-9999  Missing value
**echoSignalPower** (4-byte float, array size: nbin x nray x nscan):
Power subtracted power from echoPower to echonoise. Values are in dBm. Special values are defined as:
-9999.9 Missing value

**rangeDist** (8-byte float, array size: nbin x nray x nscan):
Distance from satellite to each range bin along each beam. Values are in m. Special values are defined as:
-9999.9 Missing value

**scanAngle** (4-byte float, array size: nray x nscan):
Angle of the beam (degrees) from nominal nadir offset about the mechanical x axis. The sign of the angle is consistent with the sensor y-axis, i.e., the angle is positive to the right of the direction of travel if the spacecraft is in normal mode.

**binRealSurfaceL1** (2-byte integer, array size: nray x nscan):
Range bin number of surface position detected by echoPower profile in DPR level 2 algorithm. Special values are defined as:
-9999 Missing value

**echoCountRealSurface** (1-byte char, array size: nray x nscan):
Echo count at a surface position (binRealSurface). Missing value = 0.

**echoPowerNoPrecipAve** (4-byte float, array size: nray x nscan):
Not used.

**echoPowerNoPrecipNum** (2-byte integer, array size: nray x nscan):
Not used.

**VERTMP** (Group)

**piaNPganal** (4-byte float, array size: piaNPGd x nray x nscan):
TBD. Values are in dB. Special values are defined as:
-9999.9 Missing value

**attenuationNPwv** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of attenuation by water vapor. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

**attenuationNPoxygen** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of attenuation by oxygen molecules. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

**attenuationNPcl** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of attenuation by cloud liquid water. Values are in dB/km. Special values
are defined as:
-9999.9 Missing value

**zFactorNPCorrected** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of reflectivity factor with attenuation correction only for non-precipitating particles. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**DSDTMP** (Group)

**particleTemperature** (4-byte float, array size: nbin x nray x nscan):
The temperature of the the hydrometeors used in calculations. Values are in K. Special values are defined as:
-9999.9 Missing value

**flagDSD** (4-byte integer, array size: nbin x nray x nscan):
A flag indicating which DSD was used. Special values are defined as:
-9999 Missing value

**attenParmAlpha** (4-byte float, array size: nbin x nray x nscan):
The attenuation parameter alpha in the attenuation-reflectivity relation k = alpha * Z **beta. Special values are defined as:
-9999.9 Missing value

**attenParmBeta** (4-byte float, array size: nbin x nray x nscan):
The attenuation parameter beta in the attenuation-reflectivity relation k = alpha * Z **beta. Special values are defined as:
-9999.9 Missing value

**reliabEpsilon** (4-byte float, array size: nray x nschan):
The reliability of epsilon. Special values are defined as:
-9999.9 Missing value

**SLVTMP** (Group)

**attenuationPrecip** (4-byte float, array size: nbin x nray x nschan):
The attenuation rate by precipitation. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

**precipWater** (4-byte float, array size: nbin x nray x nschan):
The amount of precipitable water. Values are in kg/m³. Special values are defined as:
-9999.9 Missing value
**zFactorForward1** (4-byte float, array size: nbin x nray x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorForward2** (4-byte float, array size: nbin x nray x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**C Structure Header file:**

```c
#ifndef _TK_2AKuTMPX_H_
define _TK_2AKuTMPX_H_
#endif

#ifndef _L2AKuTMPX_SLVTMP_
define _L2AKuTMPX_SLVTMP_
#endif

typedef struct {
    float attenuationPrecip[49][176];
    float precipWater[49][176];
    float zFactorForward1[49][176];
    float zFactorForward2[49][176];
} L2AKuTMPX_SLVTMP;
#endif

#ifndef _L2AKuTMPX_DSDTMP_
define _L2AKuTMPX_DSDTMP_
#endif

typedef struct {
    float particleTemperature[49][176];
    int flagDSD[49][176];
    float attenParmAlpha[49][176];
    float attenParmBeta[49][176];
    float reliabEpsilon[49];
} L2AKuTMPX_DSDTMP;
#endif

#ifndef _L2AKuTMPX_VERTMP_
define _L2AKuTMPX_VERTMP_
#endif

typedef struct {
    float piaNPganal[49][4];
    float attenuationNPwv[49][176];
```
typedef struct {
    float sidelobePower[49][176];
    float sidelobePowerError[49][176];
    short flagSidelobePower[49];
    int binRangeBottom[49];
    int binRangeTop[49];
    float echoSignalPower[49][176];
    double rangeDist[49][176];
    float scanAngle[49];
    short binRealSurfaceL1[49];
    unsigned char echoCountRealSurface[49];
    float echoPowerNoPrecipAve[49];
    short echoPowerNoPrecipNum[49];
} L2AKuTMPX_PRETMP;

#endif

#ifndef _L2AKuTMPX_RECEIVER_
#define _L2AKuTMPX_RECEIVER_

typedef struct {
    unsigned char echoCount[49][260];
    float noiseCount[49];
    short echoPower[49][260];
    short noisePower[49];
    short noiseSampleNumber[49];
    signed char echoSampleNumber[49];
    float rxAntGain[49];
    float receivedPulseWidth;
} L2AKuTMPX_RECEIVER;

#endif
#ifndef _L2AKuTMPX_TRANSMITTER_
define _L2AKuTMPX_TRANSMITTER_

typedef struct {
    float radarTransPower;
    float transPulseWidth;
    float txAntGain[49];
} L2AKuTMPX_TRANSMITTER;
#endif

#ifndef _L2AKuTMPX_VERTLOCATE_
define _L2AKuTMPX_VERTLOCATE_

typedef struct {
    short landOceanFlag[49];
    float scLocalZenith[49];
    float startBinRange[49];
    short echoHighResBinNumber[49];
    short echoLowResBinNumber[49];
    short binEllipsoid[49];
    float scRangeEllipsoid[49];
    short binDEM[49];
    float scRangeDEM[49];
    short DEMHmean[49];
    short binDEMHtop[49];
    short binDEMHbottom[49];
    short binEchoPeak[49];
    float alongTrackBeamWidth[49];
    float crossTrackBeamWidth[49];
    short mainlobeEdge[49];
    short sidelobeRange[49];
    float ellipsoidBinOffset[49];
    float rangeBinSize;
    signed char ratioLand[49];
    signed char ratioOcean[49];
    signed char ratioInLand[49];
    signed char ratioCoast[49];
} L2AKuTMPX_VERTLOCATE;
#endif

#ifndef _L2AKuTMPX_SCANSTATUS_
#define _L2AKuTMPXSCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2AKuTMPX_SCANSTATUS;

#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L2AKuTMPXFS_
#define _L2AKuTMPXFS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
float Longitude[49];
L2AKuTMPX_SCANSTATUS scanStatus;
L2AKuTMPX_VERTLOCATE VertLocate;
L2AKuTMPX_TRANSMITTER Transmitter;
L2AKuTMPX_RECEIVER Receiver;
L2AKuTMPX_PRETMP PRETMP;
L2AKuTMPX_VERTMP VERTMP;
L2AKuTMPX_DSDTMP DSDTMP;
L2AKuTMPX_SLVTMP SLVTMP;
} L2AKuTMPX_FS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L2AKuTMPX_SLVTMP/
    REAL*4 attenuationPrecip(176,49)
    REAL*4 precipWater(176,49)
    REAL*4 zFactorForward1(176,49)
    REAL*4 zFactorForward2(176,49)
END STRUCTURE

STRUCTURE /L2AKuTMPX_DSDTMP/
    REAL*4 particleTemperature(176,49)
    INTEGER*4 flagDSD(176,49)
    REAL*4 attenParmAlpha(176,49)
    REAL*4 attenParmBeta(176,49)
    REAL*4 reliabEpsilon(49)
END STRUCTURE

STRUCTURE /L2AKuTMPX_VERTMP/
    REAL*4 piaNPganal(4,49)
    REAL*4 attenuationNPwv(176,49)
    REAL*4 attenuationNPoxygen(176,49)
    REAL*4 attenuationNPcl(176,49)
    REAL*4 zFactorNPCorrected(176,49)
END STRUCTURE

STRUCTURE /L2AKuTMPX_PRETMP/
    REAL*4 sidelobePower(176,49)
    REAL*4 sidelobePowerError(176,49)
INTEGER*2 flagSidelobePower(49)
INTEGER*4 binRangeBottom(49)
INTEGER*4 binRangeTop(49)
REAL*4 echoSignalPower(176,49)
REAL*8 rangeDist(176,49)
REAL*4 scanAngle(49)
INTEGER*2 binRealSurfaceL1(49)
CHARACTER echoCountRealSurface(49)
REAL*4 echoPowerNoPrecipAve(49)
INTEGER*2 echoPowerNoPrecipNum(49)

END STRUCTURE

STRUCTURE /L2AKuTMPX_RECEIVER/
  CHARACTER echoCount(260,49)
  REAL*4 noiseCount(49)
  INTEGER*2 echoPower(260,49)
  INTEGER*2 noisePower(49)
  INTEGER*2 noiseSampleNumber(49)
  BYTE echoSampleNumber(49)
  REAL*4 rxAntGain(49)
  REAL*4 receivedPulseWidth

END STRUCTURE

STRUCTURE /L2AKuTMPX_TRANSMITTER/
  REAL*4 radarTransPower
  REAL*4 transPulseWidth
  REAL*4 txAntGain(49)

END STRUCTURE

STRUCTURE /L2AKuTMPX_VERTLOCATE/
  INTEGER*2 landOceanFlag(49)
  REAL*4 scLocalZenith(49)
  REAL*4 startBinRange(49)
  INTEGER*2 echoHighResBinNumber(49)
  INTEGER*2 echoLowResBinNumber(49)
  INTEGER*2 binEllipsoid(49)
  REAL*4 scRangeEllipsoid(49)
  INTEGER*2 binDEM(49)
  REAL*4 scRangeDEM(49)
  INTEGER*2 DEMHmean(49)
  INTEGER*2 binDEMHtop(49)
  INTEGER*2 binDEMHbottom(49)
  INTEGER*2 binEchoPeak(49)
REAL*4 alongTrackBeamWidth(49)
REAL*4 crossTrackBeamWidth(49)
INTEGER*2 mainlobeEdge(49)
INTEGER*2 sidelobeRange(49)
REAL*4 ellipsoidBinOffset(49)
REAL*4 rangeBinSize
BYTE ratioLand(49)
BYTE ratioOcean(49)
BYTE ratioInLand(49)
BYTE ratioCoast(49)
END STRUCTURE

STRUCTURE /L2AKuTMPX_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCoorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AKuTMPX_FS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
The 2AKaTMP product contains intermediate data used in the 2AKu retrieval.

Dimension definitions:

- `nscan` var: Number of scans in the granule.
- `nrays` 49: Number of angle bins in each FS scan.
- `nraysHS` 24: Number of angle bins in each HS scan.
- `nbin` 176: Number of range bins in each FS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
- `nbinln` 260: Number of L1 range bins in each FS ray. Bin interval is 125 m.
- `nbinHS` 88: Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.
- `nbinlnHS` 130: Number of L1 range bins in each HS ray. Bin interval is 250 m.
- `nNP` 4: Number of NP kinds.
- `nRScan` 4: Number of Ref Scan ID.
- `method` 6: Number of SRT methods.
- `nNode` 5: Number of binNode.
- `nDSD` 2: Number of DSD parameters. Parameters are N0 and D0.
- `LS` 2: Liquid, solid.
- `nDielec` 2: Number of dielectric constants.
- `nParmFV` 2: Number of parameters of falling velocity.
- `piaNPGd` 4: Number of parameters of piaNPGANAL.
- `two` 2: Number 2.

Figure 929 through Figure 949 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 929: Data Format Structure for 2AKaTMPX, Ka Temporary

Figure 930: Data Format Structure for 2AKaTMPX, FS
Figure 931: Data Format Structure for 2AKaTMPX, HS

Figure 932: Data Format Structure for 2AKaTMPX, FS, ScanTime
Figure 933: Data Format Structure for 2AKaTMPX, FS, scanStatus
Figure 934: Data Format Structure for 2AKaTMPX, FS, VertLocate

Figure 935: Data Format Structure for 2AKaTMPX, FS, Transmitter
Figure 936: Data Format Structure for 2AKaTMPX, FS, Receiver

Figure 937: Data Format Structure for 2AKaTMPX, FS, PRETMP

Figure 938: Data Format Structure for 2AKaTMPX, FS, VERTMP
Figure 939: Data Format Structure for 2AKaTMPX, FS, DSDTMP

Figure 940: Data Format Structure for 2AKaTMPX, FS, SLVTMP
FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

FS (Swath)

FS_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
ScanTime (Group in FS)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
Figure 943: Data Format Structure for 2AKaTMPX, HS, VertLocate

Figure 944: Data Format Structure for 2AKaTMPX, HS, Transmitter
Figure 945: Data Format Structure for 2AKaTMPX, HS, Receiver

Figure 946: Data Format Structure for 2AKaTMPX, HS, PRETMP

Figure 947: Data Format Structure for 2AKaTMPX, HS, VERTMP
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
scanStatus (Group in FS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

    Bit Meaning if bit = 1
    0   missing
    5   geoError is not zero
    6   modeStatus is not zero

dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

    Bit Meaning if bit = 1
    0   Beam matching is abnormal
    1   VPRF table is abnormal
    2   Surface table is abnormal
    3   geoWarning is not zero
    4   Operational mode is not observation mode
    5   GPS status is abnormal
    6   Spare (always 0)
    7   Check sum of L1A is abnormal

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

    Bit Meaning if bit = 1
    0   Scan is missing
    1   Science telemetry packet missing
    2   Science telemetry segment within packet missing
    3   Science telemetry other missing
    4   Housekeeping (HK) telemetry packet missing
    5   Spare (always 0)
    6   Spare (always 0)
    7   Spare (always 0)

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:
Bit Meaning if bit = 1
0 Spare (always 0)
1 SCorientation not 0 or 180
2 pointingStatus not 0
3 Non-routine limitErrorFlag
4 Non-routine operationalMode (not 1 or 11)
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A non-zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0 Latitude limit exceeded for viewed pixel locations
1 Negative scan time, invalid input
2 Error getting spacecraft attitude at scan mid-time
3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

**pointingStatus** (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
</tbody>
</table>
-8000 Non-nominal mission science orientation
-9999 Missing

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
</tbody>
</table>
7  Ku/Ka Standby VPRF Table OUT
8  Ku/Ka Standby Phase Out
9  Ku/Ka Standby Dump Out
10 Ku/Ka Standby (No Science Data)
11 Ku/Ka Independent Observation
12 Ku/Ka Independent External Calibration
13 Ku/Ka Independent Internal Calibration
14 Ku/Ka Independent SSPA Analysis
15 Ku/Ka Independent LNA Analysis
16 Ku/Ka Independent Health-Check
17 Ku/Ka Independent Standby VPRF Table OUT
18 Ku/Ka Independent Standby Phase Out
19 Ku/Ka Independent Standby Dump Out
20 Ku/Ka Independent Standby (No Science Data)

limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. LimitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

VertLocate (Group in FS)

landOceanFlag (2-byte integer, array size: nray x nscan):
Land or ocean information. The values of the flag are:
0 = Water
1 = Land
2 = Coast
3 = Water (w/ large attenuation)
4 = Land/Coast (w/ large attenuation)
Values range from 0 to 6. Special values are defined as:
-9999 Missing value

**scLocalZenith** (4-byte float, array size: nray x nscan):
The angle, in degrees, between the local zenith and the beam’s center line. The local (geodetic) zenith at the intersection of the ray and the earth ellipsoid is used. Values range from 0 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**startBinRange** (4-byte float, array size: nray x nscan):
The range to the first radar bin. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

**echoHighResBinNumber** (2-byte integer, array size: nray x nscan):
The bin number of the first radar bin with detectable echo. Values range from 0 to 260 range bin number. Special values are defined as:
-9999 Missing value

**echoLowResBinNumber** (2-byte integer, array size: nray x nscan):
The bin number of the first radar bin with detectable echo at low threshold. Values range from 0 to 260 range bin number. Special values are defined as:
-9999 Missing value

**binEllipsoid** (2-byte integer, array size: nray x nscan):
The range bin number of the earth ellipsoid. Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value

**scRangeEllipsoid** (4-byte float, array size: nray x nscan):
The spacecraft range to the Ellipsoid. Values range from 0 to 500000 m. Special values are defined as:
-9999.9 Missing value

**binDEM** (2-byte integer, array size: nray x nscan):
The bin number of the surface from digital elevation model data. Values range from 1 to 260 range bin number. Special values are defined as:
-9999 Missing value

**scRangeDEM** (4-byte float, array size: nray x nscan):
The range to the surface from digital elevation model data. Values range from 0 to 500000 m. Special values are defined as:
-9999.9 Missing value

**DEMHeight** (2-byte integer, array size: nray x nscan):
The mean of the height above Ellipsoid from digital elevation model data within the radar footprint. Values range from 0 to 9000 m. Special values are defined as:
-9999 Missing value
binDEMHtop (2-byte integer, array size: nray x nscan):
The range bin number of the maximum DEM surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11 km. Values range from 1 to 260 range bin number. Special values are defined as:
- 9999 Missing value

binDEMHbottom (2-byte integer, array size: nray x nscan):
The range bin number of the minimum DEM surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11 km. Values range from 1 to 260 range bin number. Special values are defined as:
- 9999 Missing value

binEchoPeak (2-byte integer, array size: nray x nscan):
The bin number of the peak echo in along the slant range radar profile. Values range from 1 to 260 range bin number. Special values are defined as:
- 9999 Missing value

alongTrackBeamWidth (4-byte float, array size: nray x nscan):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power in the along-track direction.

crossTrackBeamWidth (4-byte float, array size: nray x nscan):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power along the cross-track direction.

mainlobeEdge (2-byte integer, array size: nray x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

sidelobeRange (2-byte integer, array size: nray x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter position from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

ellipsoidBinOffset (4-byte float, array size: nray x nscan):
The offset of the Ellipsoid bin from the actual surface.

rangeBinSize (4-byte float, array size: nscan):
The size of the radar range bin (gate) along the direction of propagation.

ratioLand (1-byte integer, array size: nray x nscan):
The ratio of land to all other surface types within the radar footprint.

ratioOcean (1-byte integer, array size: nray x nscan):
The ratio of ocean to all other surface types within the radar footprint.

ratioInLand (1-byte integer, array size: nray x nscan):
The ratio of inland water to all other surface types within the radar footprint.

ratioCoast (1-byte integer, array size: nray x nscan):
The ratio of coast to all other surface types within the radar footprint.
Transmitter (Group in FS)

**radarTransPower** (4-byte float, array size: two x nscan):
The total (sum) power of 128 SSPA elements corrected with SSPA temperature in orbit, based on temperature test data of SSPA transmission power. Values are in dBm. Special values are defined as:
-9999.9  Missing value

**transPulseWidth** (4-byte float, array size: two x nscan):
Transmitted pulse width corrected with FCIF temperature in orbit, based on temperature test data of FCIF. Values range from 0.0000015 to 0.0000017 s. Special values are defined as:
-9999.9  Missing value

**txAntGain** (4-byte float, array size: nray x nscan):
Transmitted radar antenna effectiveness (dB).

Receiver (Group in FS)

**echoCount** (1-byte char, array size: nbinln x nray x nscan):
Special values are defined as:
255  Missing value

**noiseCount** (4-byte float, array size: nray x nscan):
Special values are defined as:
-9999.9  Missing value

**echoPower** (2-byte integer, array size: nbinln x nray x nscan):
Return power. Values in dBm are multiplied by 100 and stored in the file as a 2-byte integer. The unit in the file is thus 0.01 dBm. Bins where data is not written due to a transmission, calibration, or other problem, including an entire scan of missing bins, have the value of -32734. The range is -120 dBm to -20 dBm, which corresponds to values in the file from -12000 to -2000.

**noisePower** (2-byte integer, array size: nray x nscan):
An average of the 4 measured system noise values. Values in dBm are multiplied by 100 and stored in the file as a 2-byte integer. The unit in the file is thus 0.01 dBm. The range is -120 dBm to -20 dBm which corresponds to values in the file from -12000 to -2000. The accuracy is 0.9 dBm. Missing data are given the value of -32,734.

**noiseSampleNumber** (2-byte integer, array size: nray x nscan):
The number of noise samples used. Values range from 0 to 140 Number. Special values
are defined as:
-9999 Missing value

**echoSampleNumber** (1-byte integer, array size: nray x nscan):
Sample number of echo power stored in VPRF table. Values range from 0 to 127 Number. Special values are defined as:
-99 Missing value

**rxAntGain** (4-byte float, array size: nray x nscan):
Received radar antenna effectiveness (dB).

**receivedPulseWidth** (4-byte float, array size: two x nscan):
Received pulse width (dBm).

**PRETMP** (Group in FS)

**binRangeBottom** (4-byte integer, array size: nray x nscan):
Range bin number (1-260 for Ku, KaMS or 1-130 for KaHS) of level 1 corresponding to the bottom of level 2. Special values are defined as:
-9999 Missing value

**binRangeTop** (4-byte integer, array size: nray x nscan):
Range bin number (1-260 for Ku, KaMS or 1-130 for KaHS) of level 1 corresponding to the top of level 2. Special values are defined as:
-9999 Missing value

**echoSignalPower** (4-byte float, array size: nbin x nray x nscan):
Power subtracted power from echoPower to echonoise. Values are in dBm. Special values are defined as:
-9999.9 Missing value

**rangeDist** (8-byte float, array size: nbin x nray x nscan):
Distance from satellite to each range bin along each beam. Values are in m. Special values are defined as:
-9999.9 Missing value

**scanAngle** (4-byte float, array size: nray x nscan):
Angle of the beam (degrees) from nominal nadir offset about the mechanical x axis. The sign of the angle is consistent with the sensor y-axis, i.e., the angle is positive to the right of the direction of travel if the spacecraft is in normal mode.

**binRealSurfaceL1** (2-byte integer, array size: nray x nscan):
Range bin number of surface position detected by echoPower profile in DPR level 2 algorithm. Special values are defined as:
-9999 Missing value

**echoCountRealSurface** (1-byte char, array size: nray x nscan):
Echo count at a surface position (binRealSurface). Missing value = 0.
echoPowerNoPrecipAve (4-byte float, array size: nr x ns): Not used.

echoPowerNoPrecipNum (2-byte integer, array size: nr x ns): Not used.

sidelobePower (4-byte float, array size: nb x nr x ns): Power of sidelobe.

flagSidelobePower (2-byte integer, array size: nr x ns): A flag to show a selected table of a sidelobe database.

echoPowerLerpHS (4-byte float, array size: nb x nr x ns): TBD.

VERTMP (Group in FS)

piaNPgain (4-byte float, array size: piaNPd x nr x ns): TBD. Values are in dB. Special values are defined as:
-9999.9 Missing value

attenuationNPwv (4-byte float, array size: nb x nr x ns): Vertical profile of attenuation by water vapor. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

attenuationNPoxygen (4-byte float, array size: nb x nr x ns): Vertical profile of attenuation by oxygen molecules. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

attenuationNPcl (4-byte float, array size: nb x nr x ns): Vertical profile of attenuation by cloud liquid water. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

zFactorNPCorrected (4-byte float, array size: nb x nr x ns): Vertical profile of reflectivity factor with attenuation correction only for non-precipitating particles. Values are in dBZ. Special values are defined as:
-9999.9 Missing value

DSDTMP (Group in FS)

particleTemperature (4-byte float, array size: nb x nr x ns): The temperature of the the hydrometeors used in calculations. Values are in K. Special
values are defined as:
-9999.9  Missing value

**flagDSD** (4-byte integer, array size: nbin x nray x nscan):
A flag indicating which DSD was used. Special values are defined as:
-9999  Missing value

**attenParmAlpha** (4-byte float, array size: nbin x nray x nscan):
The attenuation parameter alpha in the attenuation-reflectivity relation \( k = \alpha \cdot Z^{**\beta} \). Special values are defined as:
-9999.9  Missing value

**attenParmBeta** (4-byte float, array size: nbin x nray x nscan):
The attenuation parameter beta in the attenuation-reflectivity relation \( k = \alpha \cdot Z^{**\beta} \). Special values are defined as:
-9999.9  Missing value

**reliabEpsilon** (4-byte float, array size: nray x nscan):
The reliability of epsilon. Special values are defined as:
-9999.9  Missing value

**SLVTMP** (Group in FS)

**attenuationPrecip** (4-byte float, array size: nbin x nray x nscan):
The attenuation rate by precipitation. Values are in dB/km. Special values are defined as:
-9999.9  Missing value

**precipWater** (4-byte float, array size: nbin x nray x nscan):
The amount of precipitable water. Values are in kg/m³. Special values are defined as:
-9999.9  Missing value

**zFactorForward1** (4-byte float, array size: nbin x nray x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9  Missing value

**zFactorForward2** (4-byte float, array size: nbin x nray x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9  Missing value

**HS** (Swath)

**HS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
ScanTime (Group in HS)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nrayHS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nrayHS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod.
Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scanStatus** (Group in HS)

**dataQuality** (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

- **Bit** Meaning if bit = 1
  - 0 missing
  - 5 geoError is not zero
  - 6 modeStatus is not zero

**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

- **Bit** Meaning if bit = 1
  - 0 Beam matching is abnormal
  - 1 VPRF table is abnormal
  - 2 Surface table is abnormal
  - 3 geoWarning is not zero
  - 4 Operational mode is not observation mode
  - 5 GPS status is abnormal
  - 6 Spare (always 0)
  - 7 Check sum of L1A is abnormal

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

- **Bit** Meaning if bit = 1
  - 0 Scan is missing
  - 1 Science telemetry packet missing
  - 2 Science telemetry segment within packet missing
  - 3 Science telemetry other missing
  - 4 Housekeeping (HK) telemetry packet missing
  - 5 Spare (always 0)
  - 6 Spare (always 0)
  - 7 Spare (always 0)
modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as
far as the status modes are concerned. modeStatus does not assess geolocation quality.
modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit =
1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other
bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits = 0 the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>
pointingStatus (2-byte integer, array size: nscan):  
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):  
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):  
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):
The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks.
limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

VertLocate (Group in HS)
landOceanFlag (2-byte integer, array size: nrayHS x nscan):
Land or ocean information. The values of the flag are:

- 0 = Water
- 1 = Land
- 2 = Coast
- 3 = Water (w/ large attenuation)
- 4 = Land/Coast (w/ large attenuation)

Values range from 0 to 6. Special values are defined as:
- -9999 Missing value

scLocalZenith (4-byte float, array size: nrayHS x nscan):
The angle, in degrees, between the local zenith and the beam’s center line. The local (geodetic) zenith at the intersection of the ray and the earth ellipsoid is used. Values range from 0 to 90 degrees. Special values are defined as:
- -9999.9 Missing value

startBinRange (4-byte float, array size: nrayHS x nscan):
The range to the first radar bin. Values range from 350000 to 500000 m. Special values are defined as:
- -9999.9 Missing value

echoHighResBinNumber (2-byte integer, array size: nrayHS x nscan):
The bin number of the first radar bin with detectable echo. Values range from 0 to 260 range bin number. Special values are defined as:
- -9999 Missing value

echoLowResBinNumber (2-byte integer, array size: nrayHS x nscan):
The bin number of the first radar bin with detectable echo at low threshold. Values range from 0 to 260 range bin number. Special values are defined as:
- -9999 Missing value

binEllipsoid (2-byte integer, array size: nrayHS x nscan):
The range bin number of the earth ellipsoid. Values range from 1 to 260 range bin number. Special values are defined as:
- -9999 Missing value

scRangeEllipsoid (4-byte float, array size: nrayHS x nscan):
The spacecraft range to the Ellipsiod. Values range from 0 to 500000 m. Special values are defined as:
- -9999.9 Missing value

binDEM (2-byte integer, array size: nrayHS x nscan):
The bin number of the surface from digital elevation model data. Values range from 1 to 260 range bin number. Special values are defined as:
- -9999 Missing value
scRangeDEM (4-byte float, array size: nrayHS x nscan):
The range to the surface from digital elevation model data. Values range from 0 to 500000 m. Special values are defined as:
-9999.9  Missing value

DEMHmean (2-byte integer, array size: nrayHS x nscan):
The mean of the height above Ellipsoid from digital elevation model data within the radar footprint. Values range from 0 to 9000 m. Special values are defined as:
-9999  Missing value

binDEMHtop (2-byte integer, array size: nrayHS x nscan):
The range bin number of the maximum DEM surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11 km. Values range from 1 to 260 range bin number. Special values are defined as:
-9999  Missing value

binDEMHbottom (2-byte integer, array size: nrayHS x nscan):
The range bin number of the minimum DEM surface elevation in a box centered on the IFOV. The first dimension is the box size, with sizes of 5 km x 5 km and 11 km x 11 km. Values range from 1 to 260 range bin number. Special values are defined as:
-9999  Missing value

binEchoPeak (2-byte integer, array size: nrayHS x nscan):
The bin number of the peak echo in along the slant range radar profile. Values range from 1 to 260 range bin number. Special values are defined as:
-9999  Missing value

alongTrackBeamWidth (4-byte float, array size: nrayHS x nscan):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power in the along-track direction.

crossTrackBeamWidth (4-byte float, array size: nrayHS x nscan):
Radar beamwidth (radians) at the point transmitted power reaches one half of peak power along the cross-track direction.

mainlobeEdge (2-byte integer, array size: nrayHS x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the edge of the clutter from the mainlobe.

sidelobeRange (2-byte integer, array size: nrayHS x nscan):
Absolute value of the difference in Range Bin Numbers between the detected surface and the clutter position from the sidelobe. A zero means no clutter indicated in this field since less than 3 bins contained significant clutter.

ellipsoidBinOffset (4-byte float, array size: nrayHS x nscan):
The offset of the Ellipsoid bin from the actual surface.

rangeBinSize (4-byte float, array size: nscan):
The size of the radar range bin (gate) along the direction of propagation.
ratioLand (1-byte integer, array size: nrayHS x nscan):
The ratio of land to all other surface types within the radar footprint.

ratioOcean (1-byte integer, array size: nrayHS x nscan):
The ratio of ocean to all other surface types within the radar footprint.

ratioInLand (1-byte integer, array size: nrayHS x nscan):
The ratio of inland water to all other surface types within the radar footprint.

ratioCoast (1-byte integer, array size: nrayHS x nscan):
The ratio of coast to all other surface types within the radar footprint.

Transmitter (Group in HS)

radarTransPower (4-byte float, array size: nscan):
The total (sum) power of 128 SSPA elements corrected with SSPA temperature in orbit, based on temperature test data of SSPA transmission power. Values are in dBm. Special values are defined as:
-9999.9 Missing value

transPulseWidth (4-byte float, array size: nscan):
Transmitted pulse width corrected with FCIF temperature in orbit, based on temperature test data of FCIF. Values range from 0.0000015 to 0.0000017 s. Special values are defined as:
-9999.9 Missing value

txAntGain (4-byte float, array size: nrayHS x nscan):
Transmitted radar antenna effectiveness (dB).

Receiver (Group in HS)

echoCount (1-byte char, array size: nbinlnHS x nrayHS x nscan):
Special values are defined as:
255 Missing value

noiseCount (4-byte float, array size: nrayHS x nscan):
Special values are defined as:
-9999.9 Missing value

echoPower (2-byte integer, array size: nbinlnHS x nrayHS x nscan):
Return power. Values in dBm are multiplied by 100 and stored in the file as a 2-byte integer. The unit in the file is thus 0.01 dBm. Bins where data is not written due to a transmission, calibration, or other problem, including an entire scan of missing bins, have the value of -32734. The range is -120 dBm to -20 dBm, which corresponds to values in the file from -12000 to -2000.
noisePower (2-byte integer, array size: nrayHS x nscan):
An average of the 4 measured system noise values. Values in dBm are multiplied by 100
and stored in the file as a 2-byte integer. The unit in the file is thus 0.01 dBm. The range
is -120 dBm to -20 dBm which corresponds to values in the file from -12000 to -2000. The
accuracy is 0.9 dBm. Missing data are given the value of -32,734.

noiseSampleNumber (2-byte integer, array size: nrayHS x nscan):
The number of noise samples used. Values range from 0 to 140 Number. Special values
are defined as:
-9999 Missing value

echoSampleNumber (1-byte integer, array size: nrayHS x nscan):
Sample number of echo power stored in VPRF table. Values range from 0 to 127 Number.
Special values are defined as:
-99 Missing value

rxAntGain (4-byte float, array size: nrayHS x nscan):
Received radar antenna effectiveness (dB).

receivedPulseWidth (4-byte float, array size: nscan):
Received pulse width (dBm).

PRETMP (Group in HS)

binRangeBottom (4-byte integer, array size: nrayHS x nscan):
Range bin number (1-260 for Ku, KaMS or 1-130 for KaHS) of level 1 corresponding to
the bottom of level 2. Special values are defined as:
-9999 Missing value

binRangeTop (4-byte integer, array size: nrayHS x nscan):
Range bin number (1-260 for Ku, KaMS or 1-130 for KaHS) of level 1 corresponding to
the top of level 2. Special values are defined as:
-9999 Missing value

echoSignalPower (4-byte float, array size: nbinHS x nrayHS x nscan):
Power subtracted power from echoPower to echonoise. Values are in dBm. Special values
are defined as:
-9999.9 Missing value

rangeDist (8-byte float, array size: nbinHS x nrayHS x nscan):
Distance from satellite to each range bin along each beam. Values are in m. Special values
are defined as:
-9999.9 Missing value

scanAngle (4-byte float, array size: nrayHS x nscan):
Angle of the beam (degrees) from nominal nadir offset about the mechanical x axis. The
sign of the angle is consistent with the sensor y-axis, i.e., the angle is positive to the right of the direction of travel if the spacecraft is in normal mode.

**binRealSurfaceL1** (2-byte integer, array size: nrayHS x nscan):
Range bin number of surface position detected by echoPower profile in DPR level 2 algorithm. Special values are defined as:
- 9999  Missing value

**echoCountRealSurface** (1-byte char, array size: nrayHS x nscan):
Echo count at a surface position (binRealSurface). Missing value = 0.

**echoPowerNoPrecipAve** (4-byte float, array size: nrayHS x nscan):
Not used.

**echoPowerNoPrecipNum** (2-byte integer, array size: nrayHS x nscan):
Not used.

**sidelobePower** (4-byte float, array size: nbinHS x nrayHS x nscan):
Power of sidelobe.

**flagSidelobePower** (2-byte integer, array size: nrayHS x nscan):
A flag to show a selected table of a sidelobe database.

**VERTMP** (Group in HS)

**piaNPganal** (4-byte float, array size: piaNPGd x nrayHS x nscan):
TBD. Values are in dB. Special values are defined as:
- 9999.9  Missing value

**attenuationNPwv** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of attenuation by water vapor. Values are in dB/km. Special values are defined as:
- 9999.9  Missing value

**attenuationNPoxygen** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of attenuation by oxygen molecules. Values are in dB/km. Special values are defined as:
- 9999.9  Missing value

**attenuationNPcl** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of attenuation by cloud liquid water. Values are in dB/km. Special values are defined as:
- 9999.9  Missing value

**zFactorNPCorrected** (4-byte float, array size: nbinHS x nrayHS x nscan):
Vertical profile of reflectivity factor with attenuation correction only for non-precipitating particles. Values are in dBZ. Special values are defined as:
- 9999.9  Missing value
**DSDTMP** (Group in HS)

**particleTemperature** (4-byte float, array size: nbinHS x nrayHS x nscan):
The temperature of the the hydrometeors used in calculations. Values are in K. Special values are defined as:
-9999.9 Missing value

**flagDSD** (4-byte integer, array size: nbinHS x nrayHS x nscan):
A flag indicating which DSD was used. Special values are defined as:
-9999 Missing value

**attenParmAlpha** (4-byte float, array size: nbinHS x nrayHS x nscan):
The attenuation parameter alpha in the attenuation-reflectivity relation \( k = \alpha \cdot Z^{\beta} \). Special values are defined as:
-9999.9 Missing value

**attenParmBeta** (4-byte float, array size: nbinHS x nrayHS x nscan):
The attenuation parameter beta in the attenuation-reflectivity relation \( k = \alpha \cdot Z^{\beta} \). Special values are defined as:
-9999.9 Missing value

**reliabEpsilon** (4-byte float, array size: nrayHS x nscan):
The reliability of epsilon. Special values are defined as:
-9999.9 Missing value

**SLVTMP** (Group in HS)

**attenuationPrecip** (4-byte float, array size: nbinHS x nrayHS x nscan):
The attenuation rate by precipitation. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

**precipWater** (4-byte float, array size: nbinHS x nrayHS x nscan):
The amount of precipitable water. Values are in \( kg/m^3 \). Special values are defined as:
-9999.9 Missing value

**zFactorForward1** (4-byte float, array size: nbinHS x nrayHS x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**zFactorForward2** (4-byte float, array size: nbinHS x nrayHS x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9 Missing value

**C Structure Header file:**
#ifndef _TK_2AKaTMPX_H_
define _TK_2AKaTMPX_H_

#ifndef _L2AKaTMPX_HS_SLVTMP_
define _L2AKaTMPX_HS_SLVTMP_

typedef struct {
    float attenuationPrecip[24][88];
    float precipWater[24][88];
    float zFactorForward1[24][88];
    float zFactorForward2[24][88];
} L2AKaTMPX_HS_SLVTMP;
#endif

#ifndef _L2AKaTMPX_HS_DSDTMP_
define _L2AKaTMPX_HS_DSDTMP_

typedef struct {
    float particleTemperature[24][88];
    int flagDSD[24][88];
    float attenParmAlpha[24][88];
    float attenParmBeta[24][88];
    float reliabEpsilon[24];
} L2AKaTMPX_HS_DSDTMP;
#endif

#ifndef _L2AKaTMPX_HS_VERTMP_
define _L2AKaTMPX_HS_VERTMP_

typedef struct {
    float piaNPganal[24][4];
    float attenuationNPwv[24][88];
    float attenuationNPoxygen[24][88];
    float attenuationNPcl[24][88];
    float zFactorNPCorrected[24][88];
} L2AKaTMPX_HS_VERTMP;
#endif

#ifndef _L2AKaTMPX_HS_PRETMP_
#define _L2AKaTMPX_HS_PRETMP_

typedef struct {
    int binRangeBottom[24];
    int binRangeTop[24];
    float echoSignalPower[24][88];
    double rangeDist[24][88];
    float scanAngle[24];
    short binRealSurfaceL1[24];
    unsigned char echoCountRealSurface[24];
    float echoPowerNoPrecipAve[24];
    short echoPowerNoPrecipNum[24];
    float sidelobePower[24][88];
    short flagSidelobePower[24];
} L2AKaTMPX_HS_PRETMP;

#endif

#ifndef _L2AKaTMPX_HS_RECEIVER_
#define _L2AKaTMPX_HS_RECEIVER_

typedef struct {
    unsigned char echoCount[24][130];
    float noiseCount[24];
    short echoPower[24][130];
    short noisePower[24];
    short noiseSampleNumber[24];
    signed char echoSampleNumber[24];
    float rxAntGain[24];
    float receivedPulseWidth;
} L2AKaTMPX_HS_RECEIVER;

#endif

#ifndef _L2AKaTMPX_HS_TRANSMITTER_
#define _L2AKaTMPX_HS_TRANSMITTER_

typedef struct {
    float radarTransPower;
    float transPulseWidth;
    float txAntGain[24];
} L2AKaTMPX_HS_TRANSMITTER;
#ifndef _L2AKaTMPX_HS_VERTLOCATE_
#define _L2AKaTMPX_HS_VERTLOCATE_

typedef struct {
    short landOceanFlag[24];
    float scLocalZenith[24];
    float startBinRange[24];
    short echoHighResBinNumber[24];
    short echoLowResBinNumber[24];
    short binEllipsoid[24];
    float scRangeEllipsoid[24];
    short binDEM[24];
    float scRangeDEM[24];
    short DEMHmean[24];
    short binDEMHtop[24];
    short binDEMBottom[24];
    short binEchoPeak[24];
    float alongTrackBeamWidth[24];
    float crossTrackBeamWidth[24];
    short mainlobeEdge[24];
    short sidelobeRange[24];
    float ellipsoidBinOffset[24];
    float rangeBinSize;
    signed char ratioLand[24];
    signed char ratioOcean[24];
    signed char ratioInLand[24];
    signed char ratioCoast[24];
} L2AKaTMPX_HS_VERTLOCATE;

#endif

#ifndef _L2AKaTMPX_HS_SCANSTATUS_
#define _L2AKaTMPX_HS_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short geoError;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
} L2AKaTMPX_HS_SCANSTATUS;

#endif
typedef struct {
    SCANTIME ScanTime;
    float Latitude[24];
    float Longitude[24];
    L2AKaTMPX_HS_SCANSTATUS scanStatus;
    L2AKaTMPX_HS_VERTLOCATE VertLocate;
    L2AKaTMPX_HS_TRANSMITTER Transmitter;
    L2AKaTMPX_HS_RECEIVER Receiver;
    L2AKaTMPX_HS_PRETMP PRETMP;
    L2AKaTMPX_HS_VERTMP VERTMP;
    L2AKaTMPX_HS_DSDTMP DSDTMP;
    L2AKaTMPX_HS_SLVTMP SLVTMP;
} L2AKaTMPX_HS;

typedef struct {
    float attenuationPrecip[49][176];
    float precipWater[49][176];
    float zFactorForward1[49][176];
    float zFactorForward2[49][176];
} L2AKaTMPX_FS_SLVTMP;

typedef struct {
    float attenuationPrecip[49][176];
    float precipWater[49][176];
    float zFactorForward1[49][176];
    float zFactorForward2[49][176];
} L2AKaTMPX_FS_DSDTMP;
#define _L2AKaTMPX_FS_DSDTMP_

typedef struct {
    float particleTemperature[49][176];
    int flagDSD[49][176];
    float attenParmAlpha[49][176];
    float attenParmBeta[49][176];
    float reliabEpsilon[49];
} L2AKaTMPX_FS_DSDTMP;

#endif

#ifndef _L2AKaTMPX_FS_VERTMP_
#define _L2AKaTMPX_FS_VERTMP_

typedef struct {
    float piaNPganal[49][4];
    float attenuationNPwv[49][176];
    float attenuationNPOxygen[49][176];
    float attenuationNPcl[49][176];
    float zFactorNPCorrected[49][176];
} L2AKaTMPX_FS_VERTMP;

#endif

#ifndef _L2AKaTMPX_FS_PRETMP_
#define _L2AKaTMPX_FS_PRETMP_

typedef struct {
    int binRangeBottom[49];
    int binRangeTop[49];
    float echoSignalPower[49][176];
    double rangeDist[49][176];
    float scanAngle[49];
    short binRealSurfaceL1[49];
    unsigned char echoCountRealSurface[49];
    float echoPowerNoPrecipAve[49];
    short echoPowerNoPrecipNum[49];
    float sidelobePower[49][176];
    short flagSidelobePower[49];
    float echoPowerLerpHS[49][176];
} L2AKaTMPX_FS_PRETMP;

#endif
#ifndef _L2AKaTMPX_FS_RECEIVER_
#define _L2AKaTMPX_FS_RECEIVER_

typedef struct {
    unsigned char echoCount[49][260];
    float noiseCount[49];
    short echoPower[49][260];
    short noisePower[49];
    short noiseSampleNumber[49];
    signed char echoSampleNumber[49];
    float rxAntGain[49];
    float receivedPulseWidth[2];
} L2AKaTMPX_FS_RECEIVER;

#endif

#ifndef _L2AKaTMPX_FS_TRANSMITTER_
#define _L2AKaTMPX_FS_TRANSMITTER_

typedef struct {
    float radarTransPower[2];
    float transPulseWidth[2];
    float txAntGain[49];
} L2AKaTMPX_FS_TRANSMITTER;

#endif

#ifndef _L2AKaTMPX_FS_VERTLOCATE_
#define _L2AKaTMPX_FS_VERTLOCATE_

typedef struct {
    short landOceanFlag[49];
    float scLocalZenith[49];
    float startBinRange[49];
    short echoHighResBinNumber[49];
    short echoLowResBinNumber[49];
    short binEllipsoid[49];
    float scRangeEllipsoid[49];
    short binDEM[49];
    float scRangeDEM[49];
    short DEMHmean[49];
} L2AKaTMPX_FS_VERTLOCATE;

#endif
short binDEMHtop[49];
short binDEMHbottom[49];
short binEchoPeak[49];
float alongTrackBeamWidth[49];
float crossTrackBeamWidth[49];
short mainlobeEdge[49];
short sidelobeRange[49];
float ellipsoidBinOffset[49];
float rangeBinSize;
signed char ratioLand[49];
signed char ratioOcean[49];
signed char ratioInLand[49];
signed char ratioCoast[49];
} L2AKaTMPX_FS_VERTLOCATE;

#endif

#ifndef _L2AKaTMPX_FS_SCANSTATUS_
define _L2AKaTMPX_FS_SCANSTATUS_

typedef struct {
  signed char dataQuality;
  signed char dataWarning;
  signed char missing;
  signed char modeStatus;
  short geoError;
  short geoWarning;
  short SCorientation;
  short pointingStatus;
  signed char acsModeMidScan;
  signed char targetSelectionMidScan;
  signed char operationalMode;
  signed char limitErrorFlag;
  double FractionalGranuleNumber;
} L2AKaTMPX_FS_SCANSTATUS;
#endif

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
  short Year;
} _SCANTIME_
#endif
signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L2AKaTMPX_FS_
#define _L2AKaTMPX_FS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2AKaTMPX_FS_SCANSTATUS scanStatus;
    L2AKaTMPX_FS_VERTLOCATE VertLocate;
    L2AKaTMPX_FS_TRANSMITTER Transmitter;
    L2AKaTMPX_FS_RECEIVER Receiver;
    L2AKaTMPX_FS_PRETMP PRETMP;
    L2AKaTMPX_FS_VERTMP VERTMP;
    L2AKaTMPX_FS_DSDTMP DSDTMP;
    L2AKaTMPX_FS_SLVTMP SLVTMP;
} L2AKaTMPX_FS;

#endif

#ifndef _L2AKaTMPX_SWATHS_
#define _L2AKaTMPX_SWATHS_

typedef struct {
    L2AKaTMPX_FS FS;
    L2AKaTMPX_HS HS;
} L2AKaTMPX_SWATHS;

#endif

#endif

Fortran Structure Header file:
STRUCTURE /L2AKaTMPX_HS_SLVTMP/
    REAL*4 attenuationPrecip(88,24)
    REAL*4 precipWater(88,24)
    REAL*4 zFactorForward1(88,24)
    REAL*4 zFactorForward2(88,24)
END STRUCTURE

STRUCTURE /L2AKaTMPX_HS_DSDTMP/
    REAL*4 particleTemperature(88,24)
    INTEGER*4 flagDSD(88,24)
    REAL*4 attenParmAlpha(88,24)
    REAL*4 attenParmBeta(88,24)
    REAL*4 reliabEpsilon(24)
END STRUCTURE

STRUCTURE /L2AKaTMPX_HS_VERTMP/
    REAL*4 piaNPganal(4,24)
    REAL*4 attenuationNPwv(88,24)
    REAL*4 attenuationNPoxygen(88,24)
    REAL*4 attenuationNPcl(88,24)
    REAL*4 zFactorNPCorrected(88,24)
END STRUCTURE

STRUCTURE /L2AKaTMPX_HS_PRETMP/
    INTEGER*4 binRangeBottom(24)
    INTEGER*4 binRangeTop(24)
    REAL*4 echoSignalPower(88,24)
    REAL*8 rangeDist(88,24)
    REAL*4 scanAngle(24)
    INTEGER*2 binRealSurfaceL1(24)
    CHARACTER echoCountRealSurface(24)
    REAL*4 echoPowerNoPrecipAve(24)
    INTEGER*2 echoPowerNoPrecipNum(24)
    REAL*4 sidelobePower(88,24)
    INTEGER*2 flagSidelobePower(24)
END STRUCTURE

STRUCTURE /L2AKaTMPX_HS_RECEIVER/
    CHARACTER echoCount(130,24)
    REAL*4 noiseCount(24)
    INTEGER*2 echoPower(130,24)
    INTEGER*2 noisePower(24)
INTEGER*2 noiseSampleNumber(24)
BYTE echoSampleNumber(24)
REAL*4 rxAntGain(24)
REAL*4 receivedPulseWidth

END STRUCTURE

STRUCTURE /L2AKaTMPX_HS_TRANSMITTER/
  REAL*4 radarTransPower
  REAL*4 transPulseWidth
  REAL*4 txAntGain(24)
END STRUCTURE

STRUCTURE /L2AKaTMPX_HS_VERTLOCATE/
  INTEGER*2 landOceanFlag(24)
  REAL*4 scLocalZenith(24)
  REAL*4 startBinRange(24)
  INTEGER*2 echoHighResBinNumber(24)
  INTEGER*2 echoLowResBinNumber(24)
  INTEGER*2 binEllipsoid(24)
  REAL*4 scRangeEllipsoid(24)
  INTEGER*2 binDEM(24)
  REAL*4 scRangeDEM(24)
  INTEGER*2 DEMHmean(24)
  INTEGER*2 binDEMHtop(24)
  INTEGER*2 binDEMHbottom(24)
  INTEGER*2 binEchoPeak(24)
  REAL*4 alongTrackBeamWidth(24)
  REAL*4 crossTrackBeamWidth(24)
  INTEGER*2 mainlobeEdge(24)
  INTEGER*2 sidelobeRange(24)
  REAL*4 ellipsoidBinOffset(24)
  REAL*4 rangeBinSize
  BYTE ratioLand(24)
  BYTE ratioOcean(24)
  BYTE ratioInLand(24)
  BYTE ratioCoast(24)
END STRUCTURE

STRUCTURE /L2AKaTMPX_HS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
BYTE limitErrorFlag
REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2AKaTMPX_HS/
RECORD /SCANTIME/ ScanTime
REAL*4 Latitude(24)
REAL*4 Longitude(24)
RECORD /L2AKaTMPX_HS_SCANSTATUS/ scanStatus
RECORD /L2AKaTMPX_HS_VERTLOCATE/ VertLocate
RECORD /L2AKaTMPX_HS_TRANSMITTER/ Transmitter
RECORD /L2AKaTMPX_HS_RECEIVER/ Receiver
RECORD /L2AKaTMPX_HS_PRETMP/ PRETMP
RECORD /L2AKaTMPX_HS_VERTMP/ VERTMP
RECORD /L2AKaTMPX_HS_DSDTMP/ DSDTMP
RECORD /L2AKaTMPX_HS_SLVTMP/ SLVTMP
END STRUCTURE

STRUCTURE /L2AKaTMPX_FS_SLVTMP/
REAL*4 attenuationPrecip(176,49)
REAL*4 precipWater(176,49)
REAL*4 zFactorForward1(176,49)
REAL*4 zFactorForward2(176,49)
END STRUCTURE

STRUCTURE /L2AKaTMPX_FS_DSDTMP/
REAL*4 particleTemperature(176,49)
INTEGER*4 flagDSD(176,49)
REAL*4 attenParmAlpha(176,49)
REAL*4 attenParmBeta(176,49)
REAL*4 reliabEpsilon(49)
END STRUCTURE

STRUCTURE /L2AKaTMPX_FS_VERTMP/
REAL*4 piaNPganal(4,49)
REAL*4 attenuationNPwv(176,49)
REAL*4 attenuationNPoxygen(176,49)
REAL*4 attenuationNPcl(176,49)
REAL*4 zFactorNPCorrected(176,49)
END STRUCTURE

STRUCTURE /L2AKaTMPX_FS_PRETMP/
  INTEGER*4 binRangeBottom(49)
  INTEGER*4 binRangeTop(49)
  REAL*4 echoSignalPower(176,49)
  REAL*8 rangeDist(176,49)
  REAL*4 scanAngle(49)
  INTEGER*2 binRealSurfaceL1(49)
  CHARACTER echoCountRealSurface(49)
  REAL*4 echoPowerNoPrecipAve(49)
  INTEGER*2 echoPowerNoPrecipNum(49)
  REAL*4 sidelobePower(176,49)
  INTEGER*2 flagSidelobePower(49)
  REAL*4 echoPowerLerpHS(176,49)
END STRUCTURE

STRUCTURE /L2AKaTMPX_FS_RECEIVER/
  CHARACTER echoCount(260,49)
  REAL*4 noiseCount(49)
  INTEGER*2 echoPower(260,49)
  INTEGER*2 noisePower(49)
  INTEGER*2 noiseSampleNumber(49)
  BYTE echoSampleNumber(49)
  REAL*4 rxAntGain(49)
  REAL*4 receivedPulseWidth(2)
END STRUCTURE

STRUCTURE /L2AKaTMPX_FS_TRANSMITTER/
  REAL*4 radarTransPower(2)
  REAL*4 transPulseWidth(2)
  REAL*4 txAntGain(49)
END STRUCTURE

STRUCTURE /L2AKaTMPX_FS_VERTLOCATE/
  INTEGER*2 landOceanFlag(49)
  REAL*4 scLocalZenith(49)
  REAL*4 startBinRange(49)
  INTEGER*2 echoHighResBinNumber(49)
  INTEGER*2 echoLowResBinNumber(49)
INTEGER*2 binEllipsoid(49)
REAL*4 scRangeEllipsoid(49)
INTEGER*2 binDEM(49)
REAL*4 scRangeDEM(49)
INTEGER*2 DEMHmean(49)
INTEGER*2 binDEMHeight(49)
INTEGER*2 binDEMHeightBottom(49)
INTEGER*2 binEchoPeak(49)
REAL*4 alongTrackBeamWidth(49)
REAL*4 crossTrackBeamWidth(49)
INTEGER*2 mainlobeEdge(49)
INTEGER*2 sidelobeRange(49)
REAL*4 ellipsoidBinOffset(49)
REAL*4 rangeBinSize
BYTE ratioLand(49)
BYTE ratioOcean(49)
BYTE ratioInLand(49)
BYTE ratioCoast(49)

END STRUCTURE

STRUCTURE /L2AKaTMPX_FS_SCANSTATUS/
    BYTE dataQuality
    BYTE dataWarning
    BYTE missing
    BYTE modeStatus
    INTEGER*2 geoError
    INTEGER*2 geoWarning
    INTEGER*2 SCorientation
    INTEGER*2 pointingStatus
    BYTE acsModeMidScan
    BYTE targetSelectionMidScan
    BYTE operationalMode
    BYTE limitErrorFlag
    REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
The 2ADPRTMP product contains intermediate data used in the 2ADPR retrieval.

Dimension definitions:
Figure 950: Data Format Structure for 2ADPRTMP, DPR Temporary

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nscan</td>
<td>Number of scans in the granule.</td>
</tr>
<tr>
<td>nray</td>
<td>Number of angle bins in each NS scan.</td>
</tr>
<tr>
<td>nrayMS</td>
<td>25 Number of angle bins in each MS scan.</td>
</tr>
<tr>
<td>nrayHS</td>
<td>24 Number of angle bins in each HS scan.</td>
</tr>
<tr>
<td>nbin</td>
<td>176 Number of range bins in each FS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.</td>
</tr>
<tr>
<td>nbinHS</td>
<td>88 Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.</td>
</tr>
<tr>
<td>nfreq</td>
<td>2 Number of frequencies.</td>
</tr>
<tr>
<td>nNP</td>
<td>4 Number of NP kinds.</td>
</tr>
<tr>
<td>nRScan</td>
<td>4 Number of Ref Scan ID.</td>
</tr>
<tr>
<td>method</td>
<td>6 Number of SRT methods.</td>
</tr>
<tr>
<td>nNode</td>
<td>5 Number of binNode.</td>
</tr>
<tr>
<td>nDSD</td>
<td>2 Number of DSD parameters. Parameters are N0 and D0.</td>
</tr>
<tr>
<td>LS</td>
<td>2 Liquid, solid.</td>
</tr>
<tr>
<td>nDielec</td>
<td>2 Number of dielectric constants.</td>
</tr>
<tr>
<td>nParmFV</td>
<td>2 Number of parameters of falling velocity.</td>
</tr>
<tr>
<td>piaNPGd</td>
<td>4 Number of parameters of piaNPGANAL.</td>
</tr>
</tbody>
</table>

Figure 950 through Figure 964 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 951: Data Format Structure for 2ADPRTMP, FS

Figure 952: Data Format Structure for 2ADPRTMP, HS

Figure 953: Data Format Structure for 2ADPRTMP, FS, ScanTime
Figure 954: Data Format Structure for 2ADPR TMP, FS, scanStatus

**scanStatus**
- `dataQuality` 1 byte Array: nfreq x nscan
- `dataWarning` 1 byte Array: nfreq x nscan
- `missing` 1 byte Array: nfreq x nscan
- `modeStatus` 1 byte Array: nfreq x nscan
- `geoError` 2 bytes Array: nfreq x nscan
- `geoWarning` 2 bytes Array: nfreq x nscan
- `SCorientation` 2 bytes Array: nscan
- `pointingStatus` 2 bytes Array: nfreq x nscan
- `acsModeMidScan` 1 byte Array: nscan
- `targetSelectionMidScan` 1 byte Array: nfreq x nscan
- `operationalMode` 1 byte Array: nfreq x nscan
- `limitErrorFlag` 1 byte Array: nfreq x nscan
- `FractionalGranuleNumber` 8 bytes Array: nscan

Figure 955: Data Format Structure for 2ADPR TMP, FS, PRETMP

**PRETMP**
- `binRealSurfaceL1` 2 bytes Array: nfreq x nray x nscan
- `echoCountRealSurface` 1 byte Array: nfreq x nray x nscan
- `echoPowerNoPrecipAve` 4 bytes Array: nfreq x nray x nscan
- `echoPowerNoPrecipNum` 2 bytes Array: nfreq x nray x nscan

Figure 956: Data Format Structure for 2ADPR TMP, FS, VERTMP

**VERTMP**
- `piaNPganal` 4 bytes Array: nfreq x piaNPGd x nray x nscan
- `attenuationNPwv` 4 bytes Array: nfreq x nbin x nray x nscan
- `attenuationNPOxygen` 4 bytes Array: nfreq x nbin x nray x nscan
- `attenuationNPCl` 4 bytes Array: nfreq x nbin x nray x nscan
- `zFactorNPCorrected` 4 bytes Array: nfreq x nbin x nray x nscan
5.65 2ADPRTMP - DPR Temporary

### DSDTMP
- **particleTemperature** 4 bytes
  - Array: nbin x nray x nscan
- **flagDSD** 4 bytes
  - Array: nbin x nray x nscan
- **attenParmAlpha** 4 bytes
  - Array: nfreq x nbin x nray x nscan
- **attenParmBeta** 4 bytes
  - Array: nfreq x nbin x nray x nscan
- **reliabEpsilon** 4 bytes
  - Array: nfreq x nray x nscan

Figure 957: Data Format Structure for 2ADPRTMP, FS, DSDTMP

### SLVTMP
- **attenuationPrecip** 4 bytes
  - Array: nfreq x nbin x nray x nscan
- **precipWater** 4 bytes
  - Array: nbin x nray x nscan
- **zFactorForward1** 4 bytes
  - Array: nfreq x nbin x nray x nscan
- **zFactorForward2** 4 bytes
  - Array: nfreq x nbin x nray x nscan

Figure 958: Data Format Structure for 2ADPRTMP, FS, SLVTMP

### ScanTime
- **Year** 2 bytes
  - Array: nscan
- **Month** 1 byte
  - Array: nscan
- **DayOfMonth** 1 byte
  - Array: nscan
- **Hour** 1 byte
  - Array: nscan
- **Minute** 1 byte
  - Array: nscan
- **Second** 1 byte
  - Array: nscan
- **MilliSecond** 2 bytes
  - Array: nscan
- **DayOfYear** 2 bytes
  - Array: nscan
- **SecondOfDay** 8 bytes
  - Array: nscan

Figure 959: Data Format Structure for 2ADPRTMP, HS, ScanTime
Figure 960: Data Format Structure for 2ADPRTMP, HS, scanStatus

Figure 961: Data Format Structure for 2ADPRTMP, HS, PRETMP
FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**FS (Swath)**

**FS_SWathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in FS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value
SecondOfDay (8-byte float, array size: nscan): A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan): The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan): The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in FS)

dataQuality (1-byte integer, array size: nfreq x nscan): A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

dataWarning (1-byte integer, array size: nfreq x nscan): Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

missing (1-byte integer, array size: nfreq x nscan): Indicates whether information is contained in the scan data. The values are:
Bit Meaning if bit = 1
0  Scan is missing
1  Science telemetry packet missing
2  Science telemetry segment within packet missing
3  Science telemetry other missing
4  Housekeeping (HK) telemetry packet missing
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

**modeStatus** (1-byte integer, array size: nfreq x nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is $2^i$). The non-routine situations follow:

Bit Meaning if bit = 1
0  Spare (always 0)
1  SC orientation not 0 or 180
2  pointingStatus not 0
3  Non-routine limitErrorFlag
4  Non-routine operationalMode (not 1 or 11)
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

**geoError** (2-byte integer, array size: nfreq x nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is $2^i$).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0  Latitude limit exceeded for viewed pixel locations
1  Negative scan time, invalid input
2  Error getting spacecraft attitude at scan mid-time
3. Error getting spacecraft ephemeris at scan mid-time
4. Invalid input non-unit ray vector for any pixel
5. Ray misses Earth for any pixel with normal pointing
6. Nadir calculation error for subsatellite position
7. Pixel count with geolocation error over threshold
8. Error in getting spacecraft attitude for any pixel
9. Error in getting spacecraft ephemeris for any pixel
10. Spare (always 0)
11. Spare (always 0)
12. Spare (always 0)
13. Spare (always 0)
14. Spare (always 0)
15. Spare (always 0)

**geoWarning** (2-byte integer, array size: nfreq x nscan):
A summary of geolocation warnings in the scan. **geoWarning** does not set a bit in **dataQuality**. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit $i = 1$ and other bits $= 0$ the unsigned integer value is $2^i$):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**SCorientation** (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector ($v$) from the satellite forward direction of motion, measured clockwise facing down. We define $v$ in the same direction as the spacecraft
axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nfreq x nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
</tbody>
</table>
3 Flight Z axis nadir, -X in flight direction
4 +90 yaw for DPR antenna pattern calibration
5 -90 yaw for DPR antenna pattern calibration
-99 Missing

**operationalMode** (1-byte integer, array size: nfreq x nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

**limitErrorFlag** (1-byte integer, array size: nfreq x nscan):

Bit flags for every ray with information about echo power limit checks.
limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

**FractionalGranuleNumber** (8-byte float, array size: nscan):

The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**PRETMP** (Group in FS)

*binRealSurfaceL1* (2-byte integer, array size: nfreq x nray x nscan):
The bin number of the actual surface computed at level-1. Special values are defined as:
-9999 Missing value

*echoCountRealSurface* (1-byte char, array size: nfreq x nray x nscan):
Echo count at a surface position (binRealSurface). Missing value = 0.

*echoPowerNoPrecipAve* (4-byte float, array size: nfreq x nray x nscan):
Not used.

*echoPowerNoPrecipNum* (2-byte integer, array size: nfreq x nray x nscan):
Not used. Values are in Count. Special values are defined as:
-9999 Missing value

**VERTMP** (Group in FS)

*piaNPganal* (4-byte float, array size: nfreq x piaNPGd x nray x nscan):
TBD. Values are in dB. Special values are defined as:
-9999.9 Missing value

*attenuationNPwv* (4-byte float, array size: nfreq x nbin x nray x nscan):
The attenuation rate due to non-precipitating water. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

*attenuationNPoxygen* (4-byte float, array size: nfreq x nbin x nray x nscan):
The attenuation rate of atmospheric oxygen. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

*attenuationNPcl* (4-byte float, array size: nfreq x nbin x nray x nscan):
The attenuation rate of cloud liquid water. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

*zFactorNPCorrected* (4-byte float, array size: nfreq x nbin x nray x nscan):
The reflectivity factor corrected for non-precipitating echo. Values are in dBZ. Special
values are defined as:
-9999.9 Missing value

**DSDTMP (Group in FS)**

**particleTemperature** (4-byte float, array size: nbin x nray x nscan):
The temperature of the hydrometeors used in calculations. Values are in K. Special values are defined as:
-9999.9 Missing value

**flagDSD** (4-byte integer, array size: nbin x nray x nscan):
A flag for the DSD model used in calculations. Special values are defined as:
-9999.9 Missing value

**attenParmAlpha** (4-byte float, array size: nfreq x nbin x nray x nscan):
The attenuation parameter alpha in the attenuation-reflectivity relation $k = \alpha \cdot Z^{\beta}$. Special values are defined as:
-9999.9 Missing value

**attenParmBeta** (4-byte float, array size: nfreq x nbin x nray x nscan):
The attenuation parameter beta in the attenuation-reflectivity relation $k = \alpha \cdot Z^{\beta}$. Special values are defined as:
-9999.9 Missing value

**reliabEpsilon** (4-byte float, array size: nfreq x nray x nscan):
The reliability of epsilon. Special values are defined as:
-9999.9 Missing value

**SLVTMP (Group in FS)**

**attenuationPrecip** (4-byte float, array size: nfreq x nbin x nray x nscan):
The attenuation rate through precipitation. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

**precipWater** (4-byte float, array size: nbin x nray x nscan):
The precipitable water in each range bin. Values are in kg/m$^3$. Special values are defined as:
-9999.9 Missing value

**zFactorForward1** (4-byte float, array size: nfreq x nbin x nray x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9 Missing value
zFactorForward2 (4-byte float, array size: nfreq x nbin x nray x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9  Missing value

HS (Swath)

HS_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in HS)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value
SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
- 9999.9  Missing value

Latitude (4-byte float, array size: nrayHS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
- 9999.9  Missing value

Longitude (4-byte float, array size: nrayHS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
- 9999.9  Missing value

scanStatus (Group in HS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is
meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit
i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>

dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:
Bit Meaning if bit = 1
0  Scan is missing
1  Science telemetry packet missing
2  Science telemetry segment within packet missing
3  Science telemetry other missing
4  Housekeeping (HK) telemetry packet missing
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

Bit Meaning if bit = 1
0  Spare (always 0)
1  SCorientation not 0 or 180
2  pointingStatus not 0
3  Non-routine limitErrorFlag
4  Non-routine operationalMode (not 1 or 11)
5  Spare (always 0)
6  Spare (always 0)
7  Spare (always 0)

geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

Bit Meaning if bit = 1
0  Latitude limit exceeded for viewed pixel locations
1  Negative scan time, invalid input
2  Error getting spacecraft attitude at scan mid-time
5.65 2ADPRTMP - DPR Temporary

3 Error getting spacecraft ephemeris at scan mid-time
4 Invalid input non-unit ray vector for any pixel
5 Ray misses Earth for any pixel with normal pointing
6 Nadir calculation error for subsatellite position
7 Pixel count with geolocation error over threshold
8 Error in getting spacecraft attitude for any pixel
9 Error in getting spacecraft ephemeris for any pixel
10 Spare (always 0)
11 Spare (always 0)
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Anomalous Time Step</td>
</tr>
<tr>
<td>5</td>
<td>GHA not calculated due to error</td>
</tr>
<tr>
<td>6</td>
<td>SunData (Group) not calculated due to error</td>
</tr>
<tr>
<td>7</td>
<td>Failure to calculate Sun in inertial coordinates</td>
</tr>
<tr>
<td>8</td>
<td>Fallback to GES ephemeris</td>
</tr>
<tr>
<td>9</td>
<td>Fallback to GEONS ephemeris</td>
</tr>
<tr>
<td>10</td>
<td>Fallback to PVT ephemeris</td>
</tr>
<tr>
<td>11</td>
<td>Fallback to OBP ephemeris</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft
axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
<tr>
<td>2</td>
<td>SUNPOINT</td>
</tr>
<tr>
<td>3</td>
<td>GSPM (Gyro-less Sun Point)</td>
</tr>
<tr>
<td>4</td>
<td>MSM (Mission Science Mode)</td>
</tr>
<tr>
<td>5</td>
<td>SLEW</td>
</tr>
<tr>
<td>6</td>
<td>DELTAH</td>
</tr>
<tr>
<td>7</td>
<td>DELTAV</td>
</tr>
<tr>
<td>-99</td>
<td>UNKNOWN -- ACS mode unavailable</td>
</tr>
</tbody>
</table>

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
</tbody>
</table>
3  Flight Z axis nadir, -X in flight direction
4  +90 yaw for DPR antenna pattern calibration
5  -90 yaw for DPR antenna pattern calibration
-99  Missing

operationalMode (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
<tr>
<td>17</td>
<td>Ku/Ka Independent Standby VPRF Table OUT</td>
</tr>
<tr>
<td>18</td>
<td>Ku/Ka Independent Standby Phase Out</td>
</tr>
<tr>
<td>19</td>
<td>Ku/Ka Independent Standby Dump Out</td>
</tr>
<tr>
<td>20</td>
<td>Ku/Ka Independent Standby (No Science Data)</td>
</tr>
</tbody>
</table>

limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. limitErrorFlag may be used in modeStatus. Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

**PRETMP** (Group in HS)

*binRealSurfaceL1* (2-byte integer, array size: nrayHS x nscan):
The bin number of the actual surface computed at level-1. Special values are defined as:
-9999 Missing value

*echoCountRealSurface* (1-byte char, array size: nrayHS x nscan):
Echo count at a surface position (binRealSurface). Missing value = 0.

*echoPowerNoPrecipAve* (4-byte float, array size: nrayHS x nscan):
Not used.

*echoPowerNoPrecipNum* (2-byte integer, array size: nrayHS x nscan):
Not used. Values are in Count. Special values are defined as:
-9999 Missing value

**VERTMP** (Group in HS)

*piaNPganal* (4-byte float, array size: piaNPGd x nrayHS x nscan):
TBD. Values are in dB. Special values are defined as:
-9999.9 Missing value

*attenuationNPwv* (4-byte float, array size: nbinHS x nrayHS x nscan):
The attenuation rate due to non-precipitating water. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

*attenuationNPOxygen* (4-byte float, array size: nbinHS x nrayHS x nscan):
The attenuation rate of atmospheric oxygen. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

*attenuationNPcl* (4-byte float, array size: nbinHS x nrayHS x nscan):
The attenuation rate of cloud liquid water. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

*zFactorNPCorrected* (4-byte float, array size: nbinHS x nrayHS x nscan):
The reflectivity factor corrected for non-precipitating echo. Values are in dBZ. Special
values are defined as:
-9999.9 Missing value

DSDTMP (Group in HS)

particleTemperature (4-byte float, array size: nbinHS x nrayHS x nscan):
The temperature of the hydrometeors used in calculations. Values are in K. Special values are defined as:
-9999.9 Missing value

flagDSD (4-byte integer, array size: nbinHS x nrayHS x nscan):
A flag for the DSD model used in calculations. Special values are defined as:
-9999 Missing value

attenParmAlpha (4-byte float, array size: nbinHS x nrayHS x nscan):
The attenuation parameter alpha in the attenuation-reflectivity relation $k = \alpha * Z^{**\beta}$. Special values are defined as:
-9999.9 Missing value

attenParmBeta (4-byte float, array size: nbinHS x nrayHS x nscan):
The attenuation parameter beta in the attenuation-reflectivity relation $k = \alpha * Z^{**\beta}$. Special values are defined as:
-9999.9 Missing value

reliabEpsilon (4-byte float, array size: nrayHS x nscan):
The reliability of epsilon. Special values are defined as:
-9999.9 Missing value

SLVTMP (Group in HS)

attenuationPrecip (4-byte float, array size: nbinHS x nrayHS x nscan):
The attenuation rate through precipitation. Values are in dB/km. Special values are defined as:
-9999.9 Missing value

precipWater (4-byte float, array size: nbinHS x nrayHS x nscan):
The precipitable water in each range bin. Values are in kg/m$^3$. Special values are defined as:
-9999.9 Missing value

zFactorForward1 (4-byte float, array size: nbinHS x nrayHS x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9 Missing value
zFactorForward2 (4-byte float, array size: nbinHS x nrayHS x nscan):
TBD Values are in dBZ. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_2ADPRTMP_H_
define _TK_2ADPRTMP_H_

#ifndef _L2ADPRTMPX_HS_SLVTMP_
define _L2ADPRTMPX_HS_SLVTMP_

typedef struct {
    float attenuationPrecip[24][88];
    float precipWater[24][88];
    float zFactorForward1[24][88];
    float zFactorForward2[24][88];
} L2ADPRTMPX_HS_SLVTMP;
#endif

#endif
#ifndef _L2ADPRTMPX_HS_DSDTMP_
define _L2ADPRTMPX_HS_DSDTMP_

typedef struct {
    float particleTemperature[24][88];
    int flagDSD[24][88];
    float attenParmAlpha[24][88];
    float attenParmBeta[24][88];
    float reliabEpsilon[24];
} L2ADPRTMPX_HS_DSDTMP;
#endif

#endif
#ifndef _L2ADPRTMPX_HS_Vertmp_
define _L2ADPRTMPX_HS_Vertmp_

typedef struct {
    float piaNPganal[24][4];
    float attenuationNPwv[24][88];
    float attenuationNPoxygen[24][88];
    float attenuationNPcl[24][88];
    float zFactorNPCorrected[24][88];
} L2ADPRTMPX_HS_Vertmp;
```
typedef struct {
    short binRealSurfaceL1[24];
    unsigned char echoCountRealSurface[24];
    float echoPowerNoPrecipAve[24];
    short echoPowerNoPrecipNum[24];
} L2ADPRTMPX_HS_PRETMP;

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2ADPRTMPX_HS_SCANSTATUS;

typedef struct {
    SCANTIME ScanTime;
    float Latitude[24];
    float Longitude[24];
L2ADPRTMPX_HS_SCANSTATUS scanStatus;
L2ADPRTMPX_HS_PRETMP PRETMP;
L2ADPRTMPX_HS_VERTMP VERTMP;
L2ADPRTMPX_HS_DSDTMP DSDTMP;
L2ADPRTMPX_HS_SLVTMP SLVTMP;
} L2ADPRTMPX_HS;

#endif

#ifndef _L2ADPRTMPX_FS_SLVTMP_
#define _L2ADPRTMPX_FS_SLVTMP_

typedef struct {
    float attenuationPrecip[49][176][2];
    float precipWater[49][176];
    float zFactorForward1[49][176][2];
    float zFactorForward2[49][176][2];
} L2ADPRTMPX_FS_SLVTMP;

#endif

#ifndef _L2ADPRTMPX_FS_DSDTMP_
#define _L2ADPRTMPX_FS_DSDTMP_

typedef struct {
    float particleTemperature[49][176];
    int flagDSD[49][176];
    float attenParmAlpha[49][176][2];
    float attenParmBeta[49][176][2];
    float reliabEpsilon[49][2];
} L2ADPRTMPX_FS_DSDTMP;

#endif

#ifndef _L2ADPRTMPX_FS_VERTMP_
#define _L2ADPRTMPX_FS_VERTMP_

typedef struct {
    float piaNPganal[49][4][2];
    float attenuationNPwv[49][176][2];
    float attenuationNPoxygen[49][176][2];
    float attenuationNPcl[49][176][2];
    float zFactorNPcorrected[49][176][2];
} L2ADPRTMPX_FS_VERTMP;

} L2ADPRTMPX_FS;
typedef struct {
    short binRealSurfaceL1[49][2];
    unsigned char echoCountRealSurface[49][2];
    float echoPowerNoPrecipAve[49][2];
    short echoPowerNoPrecipNum[49][2];
} L2ADPRTMPX_FS_PRETMP;

typedef struct {  
    signed char dataQuality[2];
    signed char dataWarning[2];
    signed char missing[2];
    signed char modeStatus[2];
    short geoError[2];
    short geoWarning[2];
    short SCorientation;
    short pointingStatus[2];
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode[2];
    signed char limitErrorFlag[2];
    double FractionalGranuleNumber;
} L2ADPRTMPX_FS_SCANSTATUS;

typedef struct {
    short Year;
    signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
}
}

 typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2ADPRTMPX_FS_SCANSTATUS scanStatus;
    L2ADPRTMPX_FS_PRETMP PRETMP;
    L2ADPRTMPX_FS_VERTMP VERTMP;
    L2ADPRTMPX_FS_DSDTMP DSDTMP;
    L2ADPRTMPX_FS_SLVTMP SLVTMP;
} L2ADPRTMPX_FS;

#end def

#ifndef _L2ADPRTMPX_SWATHS_
#define _L2ADPRTMPX_SWATHS_

typedef struct {
    L2ADPRTMPX_FS FS;
    L2ADPRTMPX_HS HS;
} L2ADPRTMPX_SWATHS;

#end def

Fortran Structure Header file:

STRUCTURE /L2ADPRTMPX_HS_SLVTMP/
    REAL*4 attenuationPrecip(88,24)
REAL*4 precipWater(88,24)
REAL*4 zFactorForward1(88,24)
REAL*4 zFactorForward2(88,24)
END STRUCTURE

STRUCTURE /L2ADPRTMPX_HS_DSDTMP/
   REAL*4 particleTemperature(88,24)
   INTEGER*4 flagDSD(88,24)
   REAL*4 attenParmAlpha(88,24)
   REAL*4 attenParmBeta(88,24)
   REAL*4 reliabEpsilon(24)
END STRUCTURE

STRUCTURE /L2ADPRTMPX_HS_VERTMP/
   REAL*4 piaNPganal(4,24)
   REAL*4 attenuationNPwv(88,24)
   REAL*4 attenuationNPoxygen(88,24)
   REAL*4 attenuationNPcl(88,24)
   REAL*4 zFactorNPCorrected(88,24)
END STRUCTURE

STRUCTURE /L2ADPRTMPX_HS_PRETMP/
   INTEGER*2 binRealSurfaceL1(24)
   CHARACTER echoCountRealSurface(24)
   REAL*4 echoPowerNoPrecipAve(24)
   INTEGER*2 echoPowerNoPrecipNum(24)
END STRUCTURE

STRUCTURE /L2ADPRTMPX_HS_SCANSTATUS/
   BYTE dataQuality
   BYTE dataWarning
   BYTE missing
   BYTE modeStatus
   INTEGER*2 geoError
   INTEGER*2 geoWarning
   INTEGER*2 SCorientation
   INTEGER*2 pointingStatus
   BYTE acsModeMidScan
   BYTE targetSelectionMidScan
   BYTE operationalMode
   BYTE limitErrorFlag
   REAL*8 FractionalGranuleNumber
END STRUCTURE
STRUCTURE /L2ADPRTMPX_HS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(24)
  REAL*4 Longitude(24)
  RECORD /L2ADPRTMPX_HS_SCANSTATUS/ scanStatus
  RECORD /L2ADPRTMPX_HS_PRETMP/ PRETMP
  RECORD /L2ADPRTMPX_HS_VERTMP/ VERTMP
  RECORD /L2ADPRTMPX_HS_DSDTMP/ DSDTMP
  RECORD /L2ADPRTMPX_HS_SLVTMP/ SLVTMP
END STRUCTURE

STRUCTURE /L2ADPRTMPX_FS_SLVTMP/
  REAL*4 attenuationPrecip(2,176,49)
  REAL*4 precipWater(176,49)
  REAL*4 zFactorForward1(2,176,49)
  REAL*4 zFactorForward2(2,176,49)
END STRUCTURE

STRUCTURE /L2ADPRTMPX_FS_DSDTMP/
  REAL*4 particleTemperature(176,49)
  INTEGER*4 flagDSD(176,49)
  REAL*4 attenParmAlpha(2,176,49)
  REAL*4 attenParmBeta(2,176,49)
  REAL*4 reliabEpsilon(2,49)
END STRUCTURE

STRUCTURE /L2ADPRTMPX_FS_VERTMP/
  REAL*4 piaNPganal(2,4,49)
  REAL*4 attenuationNPwv(2,176,49)
  REAL*4 attenuationNPoxygen(2,176,49)
  REAL*4 attenuationNPcl(2,176,49)
  REAL*4 zFactorNPCorrected(2,176,49)
END STRUCTURE

STRUCTURE /L2ADPRTMPX_FS_PRETMP/
  INTEGER*2 binRealSurfaceL1(2,49)
  CHARACTER echoCountRealSurface(2,49)
  REAL*4 echoPowerNoPrecipAve(2,49)
  INTEGER*2 echoPowerNoPrecipNum(2,49)
END STRUCTURE

STRUCTURE /L2ADPRTMPX_FS_SCANSTATUS/
BYTE dataQuality(2)
BYTE dataWarning(2)
BYTE missing(2)
BYTE modeStatus(2)
INTEGER*2 geoError(2)
INTEGER*2 geoWarning(2)
INTEGER*2 SCorientation
INTEGER*2 pointingStatus(2)
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode(2)
BYTE limitErrorFlag(2)
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay

END STRUCTURE

STRUCTURE /L2ADPRTMPX_FS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
  REAL*4 Longitude(49)
  RECORD /L2ADPRTMPX_FS_SCANSTATUS/ scanStatus
  RECORD /L2ADPRTMPX_FS_PRETMP/ PRETMP
  RECORD /L2ADPRTMPX_FS_VERTMP/ VERTMP
  RECORD /L2ADPRTMPX_FS_DSDTMP/ DSDTMP
  RECORD /L2ADPRTMPX_FS_SLVTMP/ SLVTMP

END STRUCTURE

STRUCTURE /L2ADPRTMPX_SWATHS/
  RECORD /L2ADPRTMPX_FS/ FS;
  RECORD /L2ADPRTMPX_HS/ HS;

END STRUCTURE
The 2AKuENV product contains atmospheric state information used in the DPR Ku retrieval process. This product is created by the 2AKu algorithm.

Dimension definitions:
- **nscan**: Number of scans in the granule.
- **nray**: Number of angle bins in each FS scan.
- **nrayHS**: Number of angle bins in each HS scan.
- **nbin**: Number of range bins in each NS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
- **nbinHS**: Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.
- **nNP**: Number of NP kinds.
- **nRScan**: Number of Ref Scan ID.
- **method**: Number of SRT methods.
- **nNode**: Number of binNode.
- **nDSD**: Number of DSD parameters. Parameters are N0 and D0.
- **LS**: Liquid, solid.
- **nwind**: Number of wind components: u,v.
- **nwater**: Source of water vapor data.

Figure 965 through Figure 967 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See...
Figure 966: Data Format Structure for 2AKuENVX, ScanTime

Figure 967: Data Format Structure for 2AKuENVX, VERENV
Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo (Metadata):**
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**FS (Swath)**

**SwathHeader (Metadata):**
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime (Group)**
A UTC time associated with the scan.

**Year (2-byte integer, array size: nscan):**
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month (1-byte integer, array size: nscan):**
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth (1-byte integer, array size: nscan):**
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour (1-byte integer, array size: nscan):**
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

VERENV (Group)

airPressure (4-byte float, array size: nbin x nray x nscan):
Vertical profile of air pressure inserted from the ancillary data. Values are in hPa. Special values are defined as:
-9999.9 Missing value

cloudLiquidWater (4-byte float, array size: nwater x nbin x nray x nscan):
Vertical profile of cloud liquid water. nwater = 0: a value diagnosed by the algorithm. nwater = 1: a value inserted from the ancillary data. Values are in kg/m³. Special values are defined as:
-9999.9 Missing value
skinTemperature (4-byte float, array size: nray x nscan):
Surface skin temperature inserted from the ancillary data. Values are in K. Special values are defined as:
-9999.9 Missing value

surfacePressure (4-byte float, array size: nray x nscan):
Surface pressure inserted from the ancillary data. Values are in hPa. Special values are defined as:
-9999.9 Missing value

surfaceTemperature (4-byte float, array size: nray x nscan):
Surface (2m) air temperature inserted from the ancillary data. Values are in K. Special values are defined as:
-9999.9 Missing value

surfaceWind (4-byte float, array size: nwind x nray x nscan):
Surface wind. nwind = 0: zonal direction. nwind = 1: meridional direction. Values are in m/s. Special values are defined as:
-9999.9 Missing value

waterVapor (4-byte float, array size: nwater x nbin x nray x nscan):
Vertical profile of water vapor. nwater = 0: a value diagnosed by the algorithm. nwater = 1: a value inserted from the ancillary data. Values are in kg/m³. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_2AKuENVX_H_
#define _TK_2AKuENVX_H_

#ifndef _L2AKuENVX_VERENV_
#define _L2AKuENVX_VERENV_

typedef struct {
    float airPressure[49][176];
    float cloudLiquidWater[49][176][2];
    float skinTemperature[49];
    float surfacePressure[49];
    float surfaceTemperature[49];
    float surfaceWind[49][2];
    float waterVapor[49][176][2];
} L2AKuENVX_VERENV;

#endif
```

```c
#endif
```


```c
#define _SCANTIME_

typedef struct {
  short Year;
  signed char Month;
  signed char DayOfMonth;
  signed char Hour;
  signed char Minute;
  signed char Second;
  short MilliSecond;
  short DayOfYear;
  double SecondOfDay;
} SCANTIME;

#endif

#define _L2AKuENVX_FS_

typedef struct {
  SCANTIME ScanTime;
  float Latitude[49];
  float Longitude[49];
  L2AKuENVX_VERENV VERENV;
} L2AKuENVX_FS;

#endif
#endif

Fortran Structure Header file:

```
STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2AKuENVX_FS/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L2AKuENVX_VERENV/ VERENV
END STRUCTURE

5.67 2AKaENVX - Ka environment

The 2AKaENV product contains atmospheric state information used in the 2AKu retrieval algorithm.

Dimension definitions:
- nscan  var  Number of scans in the granule.
- nray   49  Number of angle bins in each NS scan.
- nrayHS 24  Number of angle bins in each HS scan.
- nbin   176 Number of range bins in each NS and MS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
- nbinHS 88  Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.
- nNP  4  Number of NP kinds.
- nRScan 4  Number of Ref Scan ID.
- method  6  Number of SRT methods.
- nNode  5  Number of binNode.
- nDSD  2  Number of DSD parameters. Parameters are N0 and D0.
- LS  2  Liquid, solid.
- nwind  2  Number of wind components: u,v.
- nwater 2  Source of water vapor data.

Figure 968 through Figure 974 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
5.67 2AKaENVX - Ka environment

Figure 968: Data Format Structure for 2AKaENVX, Ka environment

Figure 969: Data Format Structure for 2AKaENVX, FS

Figure 970: Data Format Structure for 2AKaENVX, HS
5 STANDARD GPM PRODUCTS

ScanTime

Year 2 bytes  Array: nscan
Month 1 byte  Array: nscan
DayOfMonth 1 byte  Array: nscan
Hour 1 byte  Array: nscan
Minute 1 byte  Array: nscan
Second 1 byte  Array: nscan
MilliSecond 2 bytes  Array: nscan
DayOfYear 2 bytes  Array: nscan
SecondOfDay 8 bytes  Array: nscan

Figure 971: Data Format Structure for 2AKaENVX, FS, ScanTime

VERENV

airPressure 4 bytes  Array: nbin x nray x nscan
cloudLiquidWater 4 bytes  Array: nwater x nbin x nray x nscan
skinTemperature 4 bytes  Array: nray x nscan
surfacePressure 4 bytes  Array: nray x nscan
surfaceTemperature 4 bytes  Array: nray x nscan
surfaceWind 4 bytes  Array: nwind x nray x nscan
waterVapor 4 bytes  Array: nwater x nbin x nray x nscan

Figure 972: Data Format Structure for 2AKaENVX, FS, VERENV

ScanTime

Year 2 bytes  Array: nscan
Month 1 byte  Array: nscan
DayOfMonth 1 byte  Array: nscan
Hour 1 byte  Array: nscan
Minute 1 byte  Array: nscan
Second 1 byte  Array: nscan
MilliSecond 2 bytes  Array: nscan
DayOfYear 2 bytes  Array: nscan
SecondOfDay 8 bytes  Array: nscan

Figure 973: Data Format Structure for 2AKaENVX, HS, ScanTime
5.67 2AKaENVX - Ka environment

Figure 974: Data Format Structure for 2AKaENVX, HS, VERENV

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

FS (Swath)

FS_SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group in FS)
A UTC time associated with the scan.
**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime/sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
VERENV (Group in FS)

**airPressure** (4-byte float, array size: nbin x nray x nscan):
Vertical profile of air pressure inserted from the ancillary data. Values are in hPa. Special values are defined as:
-9999.9 Missing value

**cloudLiquidWater** (4-byte float, array size: nwater x nbin x nray x nscan):
Vertical profile of cloud liquid water. nwater = 0: a value diagnosed by the algorithm. nwater = 1: a value inserted from the ancillary data. Values are in $kg/m^3$. Special values are defined as:
-9999.9 Missing value

**skinTemperature** (4-byte float, array size: nray x nscan):
Surface skin temperature inserted from the ancillary data. Values are in K. Special values are defined as:
-9999.9 Missing value

**surfacePressure** (4-byte float, array size: nray x nscan):
Surface pressure inserted from the ancillary data. Values are in hPa. Special values are defined as:
-9999.9 Missing value

**surfaceTemperature** (4-byte float, array size: nray x nscan):
Surface (2m) air temperature inserted from the ancillary data. Values are in K. Special values are defined as:
-9999.9 Missing value

**surfaceWind** (4-byte float, array size: nwind x nray x nscan):
Surface wind. nwind = 0: zonal direction. nwind = 1: meridional direction. Values are in m/s. Special values are defined as:
-9999.9 Missing value

**waterVapor** (4-byte float, array size: nwater x nbin x nray x nscan):
Vertical profile of water vapor. nwater = 0: a value diagnosed by the algorithm. nwater = 1: a value inserted from the ancillary data. Values are in $kg/m^3$. Special values are defined as:
-9999.9 Missing value

HS (Swath)
**HS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in HS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan.scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

**Latitude** (4-byte float, array size: nrayHS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nrayHS x nsca:n):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**VERENV** (Group in HS)

**airPressure** (4-byte float, array size: nbinHS x nrayHS x nsca:n):
Vertical profile of air pressure inserted from the ancillary data. Values are in hPa. Special
values are defined as:
-9999.9 Missing value

**cloudLiquidWater** (4-byte float, array size: nwater x nbinHS x nrayHS x nsca:n):
Vertical profile of cloud liquid water. nwater = 0: a value diagnosed by the algorithm.
nwater = 1: a value inserted from the ancillary data. Values are in $kg/m^3$. Special values
are defined as:
-9999.9 Missing value

**skinTemperature** (4-byte float, array size: nrayHS x nsca:n):
Surface skin temperature inserted from the ancillary data. Values are in K. Special values
are defined as:
-9999.9 Missing value

**surfacePressure** (4-byte float, array size: nrayHS x nsca:n):
Surface pressure inserted from the ancillary data. Values are in hPa. Special values are
defined as:
-9999.9 Missing value

**surfaceTemperature** (4-byte float, array size: nrayHS x nsca:n):
Surface (2m) air temperature inserted from the ancillary data. Values are in K. Special
values are defined as:
-9999.9 Missing value

**surfaceWind** (4-byte float, array size: nwind x nrayHS x nsca:n):
Surface wind. nwind = 0: zonal direction. nwind = 1: meridional direction. Values are
in m/s. Special values are defined as:
-9999.9 Missing value

**waterVapor** (4-byte float, array size: nwater x nbinHS x nrayHS x nsca:n):
Vertical profile of water vapor. nwater = 0: a value diagnosed by the algorithm. nwater
= 1: a value inserted from the ancillary data. Values are in $kg/m^3$. Special values are
defined as:
-9999.9 Missing value
C Structure Header file:

```c
#ifndef _TK_2AKaENVX_H_
define _TK_2AKaENVX_H_

#ifndef _L2AKaENVX_HS_VERENV_
define _L2AKaENVX_HS_VERENV_

typedef struct {
    float airPressure[24][88];
    float cloudLiquidWater[24][88][2];
    float skinTemperature[24];
    float surfacePressure[24];
    float surfaceTemperature[24];
    float surfaceWind[24][2];
    float waterVapor[24][88][2];
} L2AKaENVX_HS_VERENV;

#endif

#ifndef _L2AKaENVX_HS_
define _L2AKaENVX_HS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[24];
    float Longitude[24];
    L2AKaENVX_HS_VERENV VERENV;
} L2AKaENVX_HS;

#endif

#ifndef _L2AKaENVX_FS_VERENV_
define _L2AKaENVX_FS_VERENV_

typedef struct {
    float airPressure[49][176];
    float cloudLiquidWater[49][176][2];
    float skinTemperature[49];
    float surfacePressure[49];
    float surfaceTemperature[49];
    float surfaceWind[49][2];
    float waterVapor[49][176][2];
```
typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2AKaENVX_FS_VERENV VERENV;
} L2AKaENVX_FS;

typedef struct {
    L2AKaENVX_FS FS;
    L2AKaENVX_HS HS;
} L2AKaENVX_SWATHS;
Fortran Structure Header file:

```fortran
STRUCTURE /L2A KaENVX_HS_VERENV/
  REAL*4 airPressure(88,24)
  REAL*4 cloudLiquidWater(2,88,24)
  REAL*4 skinTemperature(24)
  REAL*4 surfacePressure(24)
  REAL*4 surfaceTemperature(24)
  REAL*4 surfaceWind(2,24)
  REAL*4 waterVapor(2,88,24)
END STRUCTURE

STRUCTURE /L2A KaENVX_HS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(24)
  REAL*4 Longitude(24)
  RECORD /L2A KaENVX_HS_VERENV/ VERENV
END STRUCTURE

STRUCTURE /L2A KaENVX_FS_VERENV/
  REAL*4 airPressure(176,49)
  REAL*4 cloudLiquidWater(2,176,49)
  REAL*4 skinTemperature(49)
  REAL*4 surfacePressure(49)
  REAL*4 surfaceTemperature(49)
  REAL*4 surfaceWind(2,49)
  REAL*4 waterVapor(2,176,49)
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE
```
The 2ADPRENV product contains atmospheric state information used by the 2ADPR retrieval.

Dimension definitions:

- nscan  var  Number of scans in the granule.
- nray   49  Number of angle bins in each FS scan.
- nrayHS 24  Number of angle bins in each HS scan.
- nbin  176  Number of range bins in each FS ray. Bin interval is 125 m. 0 is at the top. 175 is the bin of the earth ellipsoid.
- nbinHS 88  Number of range bins in each HS ray. Bin interval is 250 m. 0 is at the top. 87 is the bin of the earth ellipsoid.
- nNP    4  Number of NP kinds.
- nRSScan 4  Number of Ref Scan ID.
- method 6  Number of SRT methods.
- nNode   5  Number of binNode.
- nDSD   2  Number of DSD parameters. Parameters are N0 and D0.
- LS 2  Liquid, solid.
- nwind  2  Number of wind components: u,v.
- nwater 2  Source of water vapor data.

Figure 975 through Figure 981 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information.
Figure 975: Data Format Structure for 2ADPRENVX, DPR environment

Figure 976: Data Format Structure for 2ADPRENVX, FS

Figure 977: Data Format Structure for 2ADPRENVX, HS
Figure 978: Data Format Structure for 2ADPRENVX, FS, ScanTime

- **Year**: 2 bytes, Array: nscan
- **Month**: 1 byte, Array: nscan
- **DayOfMonth**: 1 byte, Array: nscan
- **Hour**: 1 byte, Array: nscan
- **Minute**: 1 byte, Array: nscan
- **Second**: 1 byte, Array: nscan
- **MilliSecond**: 2 bytes, Array: nscan
- **DayOfYear**: 2 bytes, Array: nscan
- **SecondOfDay**: 8 bytes, Array: nscan

Figure 979: Data Format Structure for 2ADPRENVX, FS, VERENV

- **airPressure**: 4 bytes, Array: nbin x nray x nscan
- **cloudLiquidWater**: 4 bytes, Array: nwater x nbin x nray x nscan
- **skinTemperature**: 4 bytes, Array: nray x nscan
- **surfacePressure**: 4 bytes, Array: nray x nscan
- **surfaceTemperature**: 4 bytes, Array: nray x nscan
- **surfaceWind**: 4 bytes, Array: nwind x nray x nscan
- **waterVapor**: 4 bytes, Array: nwater x nbin x nray x nscan

Figure 980: Data Format Structure for 2ADPRENVX, HS, ScanTime
### Data Format Structure for 2ADPRENVX, HS, VERENV

<table>
<thead>
<tr>
<th>Variable</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>airPressure</td>
<td>4 bytes</td>
<td>Array: nbinHS x nrayHS x nscan</td>
</tr>
<tr>
<td>cloudLiquidWater</td>
<td>4 bytes</td>
<td>Array: nwater x nbinHS x nrayHS x nscan</td>
</tr>
<tr>
<td>skinTemperature</td>
<td>4 bytes</td>
<td>Array: nrayHS x nscan</td>
</tr>
<tr>
<td>surfacePressure</td>
<td>4 bytes</td>
<td>Array: nrayHS x nscan</td>
</tr>
<tr>
<td>surfaceTemperature</td>
<td>4 bytes</td>
<td>Array: nrayHS x nscan</td>
</tr>
<tr>
<td>surfaceWind</td>
<td>4 bytes</td>
<td>Array: nwind x nrayHS x nscan</td>
</tr>
<tr>
<td>waterVapor</td>
<td>4 bytes</td>
<td>Array: nwater x nbinHS x nrayHS x nscan</td>
</tr>
</tbody>
</table>

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

### FS (Swath)

**FS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in FS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- 9999 Missing value
Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

VERENV (Group in FS)
airPressure (4-byte float, array size: nbin x nray x nsan):
Vertical profile of air pressure inserted from the ancillary data. Values are in hPa. Special values are defined as:
-9999.9 Missing value

cloudLiquidWater (4-byte float, array size: nwater x nbin x nray x nsan):
Vertical profile of cloud liquid water. nwater = 0: a value diagnosed by the algorithm. nwater = 1: a value inserted from the ancillary data. Values are in $kg/m^3$. Special values are defined as:
-9999.9 Missing value

skinTemperature (4-byte float, array size: nray x nsan):
Surface skin temperature inserted from the ancillary data. Values are in K. Special values are defined as:
-9999.9 Missing value

surfacePressure (4-byte float, array size: nray x nsan):
Surface pressure inserted from the ancillary data. Values are in hPa. Special values are defined as:
-9999.9 Missing value

surfaceTemperature (4-byte float, array size: nray x nsan):
Surface (2m) air temperature inserted from the ancillary data. Values are in K. Special values are defined as:
-9999.9 Missing value

surfaceWind (4-byte float, array size: nwind x nray x nsan):
Surface wind. nwind = 0: zonal direction. nwind = 1: meridional direction. Values are in m/s. Special values are defined as:
-9999.9 Missing value

waterVapor (4-byte float, array size: nwater x nbin x nray x nsan):
Vertical profile of water vapor. nwater = 0: a value diagnosed by the algorithm. nwater = 1: a value inserted from the ancillary data. Values are in $kg/m^3$. Special values are defined as:
-9999.9 Missing value

**HS** (Swath)

**HS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
ScanTime (Group in HS)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nrayHS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nrayHS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

VERENV (Group in HS)

airPressure (4-byte float, array size: nbinHS x nrayHS x nsan):
Vertical profile of air pressure inserted from the ancillary data. Values are in hPa. Special values are defined as:
-9999.9 Missing value

cloudLiquidWater (4-byte float, array size: nwater x nbinHS x nrayHS x nsan):
Vertical profile of cloud liquid water. nwater = 0: a value diagnosed by the algorithm. nwater = 1: a value inserted from the ancillary data. Values are in $kg/m^3$. Special values are defined as:
-9999.9 Missing value

skinTemperature (4-byte float, array size: nrayHS x nsan):
Surface skin temperature inserted from the ancillary data. Values are in K. Special values are defined as:
-9999.9 Missing value

surfacePressure (4-byte float, array size: nrayHS x nsan):
Surface pressure inserted from the ancillary data. Values are in hPa. Special values are defined as:
-9999.9 Missing value

surfaceTemperature (4-byte float, array size: nrayHS x nsan):
Surface (2m) air temperature inserted from the ancillary data. Values are in K. Special values are defined as:
-9999.9 Missing value

surfaceWind (4-byte float, array size: nwind x nrayHS x nsan):
Surface wind. nwind = 0: zonal direction. nwind = 1: meridional direction. Values are in m/s. Special values are defined as:
-9999.9 Missing value

waterVapor (4-byte float, array size: nwater x nbinHS x nrayHS x nsan):
Vertical profile of water vapor. nwater = 0: a value diagnosed by the algorithm. nwater = 1: a value inserted from the ancillary data. Values are in $kg/m^3$. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_2ADPRENVX_H_
#define _TK_2ADPRENVX_H_

#define _TK_2ADPRENVX_H_
```
```c
#ifndef _L2ADPRENVX_HS_VERENV_
define _L2ADPRENVX_HS_VERENV_

typedef struct {
    float airPressure[24][88];
    float cloudLiquidWater[24][88][2];
    float skinTemperature[24];
    float surfacePressure[24];
    float surfaceTemperature[24];
    float surfaceWind[24][2];
    float waterVapor[24][88][2];
} L2ADPRENVX_HS_VERENV;
#endif

#ifndef _L2ADPRENVX_HS_
define _L2ADPRENVX_HS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[24];
    float Longitude[24];
    L2ADPRENVX_HS_VERENV VERENV;
} L2ADPRENVX_HS;
#endif

#ifndef _L2ADPRENVX_FS_VERENV_
define _L2ADPRENVX_FS_VERENV_

typedef struct {
    float airPressure[49][176];
    float cloudLiquidWater[49][176][2];
    float skinTemperature[49];
    float surfacePressure[49];
    float surfaceTemperature[49];
    float surfaceWind[49][2];
    float waterVapor[49][176][2];
} L2ADPRENVX_FS_VERENV;
#endif
```
```c
#include <stdio.h>

#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;

#endif

define _L2ADPRENVX_FS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2ADPRENVX_FS_VERENV VERENV;
} L2ADPRENVX_FS;
#endif

define _L2ADPRENVX_SWATHS_

typedef struct {
    L2ADPRENVX_FS FS;
    L2ADPRENVX_HS HS;
} L2ADPRENVX_SWATHS;
#endif

Fortran Structure Header file:
```
STRUCTURE /L2ADPRENVX_HS_VERENV/
  REAL*4 airPressure(88,24)
  REAL*4 cloudLiquidWater(2,88,24)
  REAL*4 skinTemperature(24)
  REAL*4 surfacePressure(24)
  REAL*4 surfaceTemperature(24)
  REAL*4 surfaceWind(2,24)
  REAL*4 waterVapor(2,88,24)
END STRUCTURE

STRUCTURE /L2ADPRENVX_HS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(24)
  REAL*4 Longitude(24)
  RECORD /L2ADPRENVX_HS_VERENV/ VERENV
END STRUCTURE

STRUCTURE /L2ADPRENVX_FS_VERENV/
  REAL*4 airPressure(176,49)
  REAL*4 cloudLiquidWater(2,176,49)
  REAL*4 skinTemperature(49)
  REAL*4 surfacePressure(49)
  REAL*4 surfaceTemperature(49)
  REAL*4 surfaceWind(2,49)
  REAL*4 waterVapor(2,176,49)
END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2ADPRENVX_FS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
  REAL*4 Longitude(49)
5.69 3DPRX - DPR Full Product

3DPR, ”DPR Full Product”, computes statistics of the DPR measurements at both a low horizontal resolution (G1, 5° x 5° latitude/longitude) and a high horizontal resolution (G2, 0.25° x 0.25° latitude/longitude). The product can be monthly or daily.

Histograms have the following category thresholds, where
\[ \text{histbin}(i) = \text{cat}(i) \text{ less than } x \text{ less than or equal to } \text{cat}(i+1) \]

\[ \text{cat rain} = \left[ 0.01, 0.10, 0.52, 2.75, 14.40, 75.44 \right] \text{ mm/h (logarithmic steps)} \]

\[ \text{cat Z} = \left[ 6.0, 18.0, 30.0, 42.0, 54.0 \right] \text{ dBZ} \]

\[ \text{cat integratedWater} = \left[ 200.0, 1400.0, 3800.0, 5000.0 \right] \text{ kg/m}^2 \]

\[ \text{cat bbhgt} = \left[ 10.0, 250.0, 1750.0, 3250.0 \right] \text{ meters} \]
5.69 3DPRX - DPR Full Product

4750.0, 5000.0, 5250.0, 5500.0, 5750.0, 6000.0, 6250.0, 6500.0, 6750.0, 7000.0, 7500.0, 20000.0 ],

cat bbwdth = [ 0.0, ! meters
125.0, 250.0, 375.0, 500.0, 625.0, 750.0, 875.0, 1000.0, 1125.0, 1250.0, 1375.0, 1500.0, 1625.0, 1750.0, 1875.0, 2000.0, 2125.0, 2250.0, 2375.0, 2500.0, 2625.0, 2750.0, 2875.0, 3000.0, 3125.0, 3250.0, 3375.0, 3500.0, 3625.0, 3750.0 ],

cat stormh = 1000.0*[ 0.01, ! km (convert m > km)
0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0, 8.5, 9.0, 9.5, 10.0, 10.5, 11.0, 11.5, 12.0, 12.5, 13.0, 14.0, 15.0, 16.0, 20.0 ],

cat epsilon = [ 0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0 ],

cat nubf = [ 1.0, 1.05, 1.1, 1.15, 1.2, 1.25, 1.3, 1.35, 1.4, 1.45, 1.5, 1.55, 1.6, 1.65, 1.7, 1.75, 1.8, 1.85, 1.9, 1.95, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0 ],

cat pia = [ 0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.8, 1.0, 1.2, 1.4, 1.6, 1.8, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 7.0, 8.0, 9.0, 10.0, 15.0, 20.0, 25.0, 30.0, 100.0 ],

cat dBNw = [ 0.1, 1.0, 2.0, 4.0, 6.0, 8.0, 10.0, 12.0, 14.0, 16.0, 18.0, 20.0, 22.0, 24.0, 26.0, 28.0, 30.0, 32.0, 34.0, 36.0, 38.0, 40.0, 42.0, 44.0, 46.0 ],
48.0, 50.0, 52.0, 54.0, 56.0, 60.0],

cat Dm = [ 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0, 4.0 ]

Dimension definitions:

ltL 28 Number of low resolution 5° grid intervals of latitude from 70°S to 70°N.

lnL 72 Number of low resolution 5° grid intervals of longitude from 180°W to 180°E.

ltH 536 Number of high resolution 0.25° grid intervals of latitude from 67°S to 67°N.

lnH 1440 Number of high resolution 0.25° grid intervals of longitude from 180°W to 180°E.

chn 7 TEST value of 7. Number of channels: KuFS 49, KaMS 25, KaHS 24, DPRMS 25, KuMS 25, KaFS 49, DPRFS 49

inst 4 Number of instruments: Ku, Ka, KaHS, KuMS.

hgt 5 Number of heights above the earth ellipsoid: 2, 4, 6, 10, and 15 km.

tim 24 Number of hours (local time).

ang 7 Number of angles. The meaning of ang is different for each channel. For Ku channel all indeces are used with the meaning 0, 1, 2,...,6 = angle bins 24, (20,28), (16,32), (12,36), (8,40), (3,44), and (0,48). For Ka channel 4 indeces are used with the meaning 0, 1, 2, 3 = angle bins 12, (8,16), (4,20), and (0,24). For KaHS channel 4 indeces are used with the meaning 0, 1, 2, 3 = angle bins (11,2), (7,16), (3,20), and (0,23).

rt 3 Number of rain types: stratiform, convective, all.

st 3 Number of surface types: ocean, land, all.

bin 30 Number of bins in histogram. The thresholds are different for different variables. See the introduction to this algorithm.

Figure 982 through Figure 1072 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 982: Data Format Structure for 3DPRX, DPR Full Product
Figure 983: Data Format Structure for 3DPRX, G1, G1
continued from last figure

- dBNW
- epsilonDPR
- epsilon
- zeta
- piaHB
- piaHybrid
- piaHybridDPR
- piaSRT
- piaSRTdpr
- piaFinal
- piaFinalDPR
- piaFinalSubset
- piaFinalDPRsubset
- heightBB
- heightBBnadir
- BBwidthNadir
- heightStormTop
- BBwidth
- observationCounts
- precipRateLocalTime
- DFRmNearSurface
- DFRNearSurface
- precipRateNearSurfaceUnconditional
  Array: RL x lN x chn
- precipProbabilityNearSurface
  Array: RL x lN x chn

continued on next figure

Figure 984: Data Format Structure for 3DPRX, G1, G1
continued from last figure

Figure 985: Data Format Structure for 3DPRX, G1

```
G1
```

```
G2_GridHeader          Metadata
  precipRate            Group
  rainRate              Group
  snowRate              Group
  flagHeavyIcePrecip    Group
  mixedPhRate           Group
  precipRateESurface    Group
  precipRateESurface2   Group
  precipRateNearSurface Group
  rainRateNearSurface   Group
  snowRateNearSurface   Group
  mixedPhRateNearSurface Group
  precipWaterIntegrated Group
  precipIceIntegrated   Group
  precipRateAve24       Group
  zFactorCorrected      Group
  zFactorCorrectedESurface Group
  zFactorCorrectedNearSurface Group
  zFactorCorrectedDPR   Group
  zFactorCorrectedESurfaceDPR Group
  zFactorCorrectedNearSurfaceDPR Group
  zFactorMeasured       Group
  dm                    Group
```

continued on next figure

```
G2
```

```
G2
```

Figure 986: Data Format Structure for 3DPRX, G2, G2
continued from last figure

- Group: dBNe
- Group: epsilonDPR
- Group: epsilon
- Group: eta
- Group: piaHB
- Group: piaHybrid
- Group: piaHybridDPR
- Group: piaSRT
- Group: piaSRTDPR
- Group: piaFinal
- Group: piaFinalDPR
- Group: heightBB
- Group: heightStormTop
- Group: BBwidth
- Group: observationCounts
- Group: DFRmNearSurface
- Group: DFRNearSurface
- Array: Array: ltH x lnH x chn
- Array: Array: ltH x lnH x chn

Figure 987: Data Format Structure for 3DPRX, G2

- Array: ltL x lnL x chn x hgt x rt x st
- Array: ltL x lnL x chn x hgt x rt x st
- Array: ltL x lnL x chn x hgt x rt x st
- Array: ltL x lnL x chn x hgt x rt x st x bin

Figure 988: Data Format Structure for 3DPRX, G1, precipRate
Figure 989: Data Format Structure for 3DPRX, G1, rainRate

Figure 990: Data Format Structure for 3DPRX, G1, snowRate

Figure 991: Data Format Structure for 3DPRX, G1, flagHeavyIcePrecip

Figure 992: Data Format Structure for 3DPRX, G1, mixedPhRate

Figure 993: Data Format Structure for 3DPRX, G1, precipRateESurface
precipRateESurface2
- count: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st\)
- mean: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st\)
- stdev: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st\)
- hist: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st \times bin\)

Figure 994: Data Format Structure for 3DPRX, G1, precipRateESurface2

precipRateNearSurface
- count: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st\)
- mean: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st\)
- stdev: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st\)
- hist: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st \times bin\)

Figure 995: Data Format Structure for 3DPRX, G1, precipRateNearSurface

rainRateNearSurface
- count: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st\)
- mean: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st\)
- stdev: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st\)
- hist: 4 bytes, Array: \(ltL \times lnL \times chn \times rt \times st \times bin\)

Figure 996: Data Format Structure for 3DPRX, G1, rainRateNearSurface
snowRateNearSurface

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

```
Array: ltL x lnL x chn x rt x st
```

mixedPhRateNearSurface

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

```
Array: ltL x lnL x chn x rt x st
```

precipWaterIntegrated

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

```
Array: ltL x lnL x chn x rt x st
```

precipIceIntegrated

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

```
Array: ltL x lnL x chn x rt x st x bin
```

precipRateAve24

- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
- hist: 4 bytes

```
Array: ltL x lnL x chn x rt x st x bin
```

Figure 997: Data Format Structure for 3DPRX, G1, snowRateNearSurface

Figure 998: Data Format Structure for 3DPRX, G1, mixedPhRateNearSurface

Figure 999: Data Format Structure for 3DPRX, G1, precipWaterIntegrated

Figure 1000: Data Format Structure for 3DPRX, G1, precipIceIntegrated

Figure 1001: Data Format Structure for 3DPRX, G1, precipRateAve24
**Figure 1002: Data Format Structure for 3DPRX, G1, zFactorCorrected**

- **zFactorCorrected**
  - count: 4 bytes
  - mean: 4 bytes
  - stddev: 4 bytes
  - hist: 4 bytes

  Array: ltL x lnL x chn x hgt x rt x st

**Figure 1003: Data Format Structure for 3DPRX, G1, zFactorCorrectedESurface**

- **zFactorCorrectedESurface**
  - count: 4 bytes
  - mean: 4 bytes
  - stddev: 4 bytes
  - hist: 4 bytes

  Array: ltL x lnL x chn x x rt x st

**Figure 1004: Data Format Structure for 3DPRX, G1, zFactorCorrectedNearSurface**

- **zFactorCorrectedNearSurface**
  - count: 4 bytes
  - mean: 4 bytes
  - stddev: 4 bytes
  - hist: 4 bytes

  Array: ltL x lnL x chn x rt x st

**Figure 1005: Data Format Structure for 3DPRX, G1, zFactorMeasuredNearSurface**

- **zFactorMeasuredNearSurface**
  - count: 4 bytes
  - mean: 4 bytes
  - stddev: 4 bytes
  - hist: 4 bytes

  Array: ltL x lnL x chn x rt x st

**Figure 1006: Data Format Structure for 3DPRX, G1, zFactorCorrectedDPR**

- **zFactorCorrectedDPR**
  - count: 4 bytes
  - mean: 4 bytes
  - stddev: 4 bytes
  - hist: 4 bytes

  Array: ltL x lnL x inst x hgt x rt x st
Figure 1007: Data Format Structure for 3DPRX, G1, zFactorCorrectedESurfaceDPR

Figure 1008: Data Format Structure for 3DPRX, G1, zFactorCorrectedNearSurfaceDPR

Figure 1009: Data Format Structure for 3DPRX, G1, zFactorMeasured

Figure 1010: Data Format Structure for 3DPRX, G1, dm

Figure 1011: Data Format Structure for 3DPRX, G1, dBNw
Figure 1012: Data Format Structure for 3DPRX, G1, epsilonDPR

**epsilonDPR**
- count 4 bytes  Array: ltL x lnL x inst x hgt x rt x st
- mean 4 bytes  Array: ltL x lnL x inst x hgt x rt x st
- stdev 4 bytes  Array: ltL x lnL x inst x hgt x rt x st
- hist 4 bytes  Array: ltL x lnL x inst x hgt x rt x st x bin

Figure 1013: Data Format Structure for 3DPRX, G1, epsilon

**epsilon**
- count 4 bytes  Array: ltL x lnL x chn x rt x st
- mean 4 bytes  Array: ltL x lnL x chn x rt x st
- stdev 4 bytes  Array: ltL x lnL x chn x rt x st
- hist 4 bytes  Array: ltL x lnL x chn x rt x st x bin

Figure 1014: Data Format Structure for 3DPRX, G1, zeta

**zeta**
- count 4 bytes  Array: ltL x lnL x chn x ang x rt x st
- mean 4 bytes  Array: ltL x lnL x chn x ang x rt x st
- stdev 4 bytes  Array: ltL x lnL x chn x ang x rt x st
- hist 4 bytes  Array: ltL x lnL x chn x ang x rt x st x bin

Figure 1015: Data Format Structure for 3DPRX, G1, piaHB

**piaHB**
- count 4 bytes  Array: ltL x lnL x chn x ang x rt x st
- mean 4 bytes  Array: ltL x lnL x chn x ang x rt x st
- stdev 4 bytes  Array: ltL x lnL x chn x ang x rt x st
- hist 4 bytes  Array: ltL x lnL x chn x ang x rt x st x bin

Figure 1016: Data Format Structure for 3DPRX, G1, piaHybrid
Figure 1017: Data Format Structure for 3DPRX, G1, piaHybridDPR

Figure 1018: Data Format Structure for 3DPRX, G1, piaSRT

Figure 1019: Data Format Structure for 3DPRX, G1, piaSRTdp

Figure 1020: Data Format Structure for 3DPRX, G1, piaFinal

Figure 1021: Data Format Structure for 3DPRX, G1, piaFinalDPR
Figure 1022: Data Format Structure for 3DPRX, G1, piaFinalSubset

Figure 1023: Data Format Structure for 3DPRX, G1, piaFinalDPRsubset

Figure 1024: Data Format Structure for 3DPRX, G1, heightBB

Figure 1025: Data Format Structure for 3DPRX, G1, heightBBnadir

Figure 1026: Data Format Structure for 3DPRX, G1, BBwidthNadir
### heightStormTop

<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x rt x st</td>
</tr>
<tr>
<td>mean</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x rt x st</td>
</tr>
<tr>
<td>stdev</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x rt x st</td>
</tr>
<tr>
<td>hist</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x rt x st x bin</td>
</tr>
</tbody>
</table>

**Figure 1027:** Data Format Structure for 3DPRX, G1, heightStormTop

### BBwidth

<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x rt x st</td>
</tr>
<tr>
<td>mean</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x rt x st</td>
</tr>
<tr>
<td>stdev</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x rt x st</td>
</tr>
<tr>
<td>hist</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x rt x st x bin</td>
</tr>
</tbody>
</table>

**Figure 1028:** Data Format Structure for 3DPRX, G1, BBwidth

### observationCounts

<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>total</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x st</td>
</tr>
<tr>
<td>localTime</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x tim x st</td>
</tr>
<tr>
<td>pia</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x ang x st</td>
</tr>
<tr>
<td>shallowRain</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x st</td>
</tr>
</tbody>
</table>

**Figure 1029:** Data Format Structure for 3DPRX, G1, observationCounts

### precipRateLocalTime

<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x tim x st</td>
</tr>
<tr>
<td>mean</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x tim x st</td>
</tr>
<tr>
<td>stdev</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x chn x tim x st</td>
</tr>
</tbody>
</table>

**Figure 1030:** Data Format Structure for 3DPRX, G1, precipRateLocalTime

### DFRmNearSurface

<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x rt x st</td>
</tr>
<tr>
<td>mean</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x rt x st</td>
</tr>
<tr>
<td>stdev</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x rt x st</td>
</tr>
<tr>
<td>hist</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x rt x st x bin</td>
</tr>
</tbody>
</table>

**Figure 1031:** Data Format Structure for 3DPRX, G1, DFRmNearSurface

### DFRNearSurface

<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x rt x st</td>
</tr>
<tr>
<td>mean</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x rt x st</td>
</tr>
<tr>
<td>stdev</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x rt x st</td>
</tr>
<tr>
<td>hist</td>
<td>4 bytes</td>
<td>Array: ltL x lnL x rt x st x bin</td>
</tr>
</tbody>
</table>

**Figure 1032:** Data Format Structure for 3DPRX, G1, DFRNearSurface

---

5 STANDARD GPM PRODUCTS
Figure 1033: Data Format Structure for 3DPRX, G2, precipRate

Figure 1034: Data Format Structure for 3DPRX, G2, rainRate

Figure 1035: Data Format Structure for 3DPRX, G2, snowRate

Figure 1036: Data Format Structure for 3DPRX, G2, flagHeavyIcePrecip
Figure 1037: Data Format Structure for 3DPRX, G2, mixedPhRate

```
+----------------+           +----------------+           +----------------+
| mixedPhRate    |           | count          |           | 4 bytes         |
|                |           | mean           |           |                  |
|                |           | stdev          |           |                  |
|                |           | Array: ltH x lnH x chn x hgt x rt |
```

Figure 1038: Data Format Structure for 3DPRX, G2, precipRateESurface

```
+----------------+           +----------------+           +----------------+
| precipRateESurface |          | count          |           | 4 bytes         |
|                   |           | mean           |           |                  |
|                   |           | stdev          |           |                  |
|                   |           | Array: ltH x lnH x chn x rt |
```

Figure 1039: Data Format Structure for 3DPRX, G2, precipRateESurface2

```
+----------------+           +----------------+           +----------------+
| precipRateESurface2 |         | count          |           | 4 bytes         |
|                    |           | mean           |           |                  |
|                    |           | stdev          |           |                  |
|                    |           | Array: ltH x lnH x chn x rt |
```

Figure 1040: Data Format Structure for 3DPRX, G2, precipRateNearSurface

```
+----------------+           +----------------+           +----------------+
| precipRateNearSurface |        | count          |           | 4 bytes         |
|                       |           | mean           |           |                  |
|                       |           | stdev          |           |                  |
|                       |           | Array: ltH x lnH x chn x rt |
```

Figure 1041: Data Format Structure for 3DPRX, G2, rainRateNearSurface

```
+----------------+           +----------------+           +----------------+
| rainRateNearSurface |         | count          |           | 4 bytes         |
|                     |           | mean           |           |                  |
|                     |           | stdev          |           |                  |
|                     |           | Array: ltH x lnH x chn x rt |
```

Figure 1042: Data Format Structure for 3DPRX, G2, snowRateNearSurface

```
+----------------+           +----------------+           +----------------+
| snowRateNearSurface |         | count          |           | 4 bytes         |
|                     |           | mean           |           |                  |
|                     |           | stdev          |           |                  |
|                     |           | Array: ltH x lnH x chn x rt |
```

Figure 1043: Data Format Structure for 3DPRX, G2, mixedPhRateNearSurface

```
+----------------+           +----------------+           +----------------+
| mixedPhRateNearSurface |       | count          |           | 4 bytes         |
|                        |           | mean           |           |                  |
|                        |           | stdev          |           |                  |
|                        |           | Array: ltH x lnH x chn x rt |
```
Figure 1044: Data Format Structure for 3DPRX, G2, precipWaterIntegrated

Figure 1045: Data Format Structure for 3DPRX, G2, precipIceIntegrated

Figure 1046: Data Format Structure for 3DPRX, G2, precipRateAve24
Figure 1054: Data Format Structure for 3DPRX, G2, zFactorMeasured

Figure 1055: Data Format Structure for 3DPRX, G2, dm

Figure 1056: Data Format Structure for 3DPRX, G2, dBm
Figure 1057: Data Format Structure for 3DPRX, G2, epsilonDPR

epsilonDPR
- count 4 bytes: Array: ltH x lnH x inst x hgt x rt
- mean 4 bytes: Array: ltH x lnH x inst x hgt x rt
- stdev 4 bytes: Array: ltH x lnH x inst x hgt x rt

Figure 1058: Data Format Structure for 3DPRX, G2, epsilon

epsilon
- count 4 bytes: Array: ltH x lnH x chn x rt
- mean 4 bytes: Array: ltH x lnH x chn x rt
- stdev 4 bytes: Array: ltH x lnH x chn x rt

Figure 1059: Data Format Structure for 3DPRX, G2, zeta

zeta
- count 4 bytes: Array: ltH x lnH x chn x ang x rt
- mean 4 bytes: Array: ltH x lnH x chn x ang x rt
- stdev 4 bytes: Array: ltH x lnH x chn x ang x rt

Figure 1060: Data Format Structure for 3DPRX, G2, piaHB

piaHB
- count 4 bytes: Array: ltH x lnH x chn x ang x rt
- mean 4 bytes: Array: ltH x lnH x chn x ang x rt
- stdev 4 bytes: Array: ltH x lnH x chn x ang x rt

Figure 1061: Data Format Structure for 3DPRX, G2, piaHybrid

piaHybrid
- count 4 bytes: Array: ltH x lnH x chn x ang x rt
- mean 4 bytes: Array: ltH x lnH x chn x ang x rt
- stdev 4 bytes: Array: ltH x lnH x chn x ang x rt

Figure 1062: Data Format Structure for 3DPRX, G2, piaHybridDPR

piaHybridDPR
- count 4 bytes: Array: ltH x lnH x inst x ang x rt
- mean 4 bytes: Array: ltH x lnH x inst x ang x rt
- stdev 4 bytes: Array: ltH x lnH x inst x ang x rt

Figure 1063: Data Format Structure for 3DPRX, G2, piaSRT

piaSRT
- count 4 bytes: Array: ltH x lnH x chn x ang x rt
- mean 4 bytes: Array: ltH x lnH x chn x ang x rt
- stdev 4 bytes: Array: ltH x lnH x chn x ang x rt
Figure 1064: Data Format Structure for 3DPRX, G2, piaSRTdpr

Figure 1065: Data Format Structure for 3DPRX, G2, piaFinal

Figure 1066: Data Format Structure for 3DPRX, G2, piaFinalDPR
**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputFileNames** (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

**InputAlgorithmVersions** (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

**InputGenerationDateTimes** (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

---

**Grids** (Group)
**5.69 3DPRX - DPR Full Product**

**observationCounts**
- **total** 4 bytes Array: ltH x lnH x chn
- **pia** 4 bytes Array: ltH x lnH x chn x ang
- **shallowRain** 4 bytes Array: ltH x lnH x chn

Figure 1070: Data Format Structure for 3DPRX, G2, observationCounts

**DFRmNearSurface**
- **count** 4 bytes Array: ltH x lnH x rt
- **mean** 4 bytes Array: ltH x lnH x rt
- **stdev** 4 bytes Array: ltH x lnH x rt

Figure 1071: Data Format Structure for 3DPRX, G2, DFRmNearSurface

**G1 (Grid)**

**G1_GridHeader (Metadata):**
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**precipRate (Group in G1)**
Conditional Precipitation Rate.

- **count** (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
  Count. Special values are defined as:
  -9999  Missing value

- **mean** (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
  mean. Special values are defined as:
  -9999.9  Missing value

- **stdev** (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
  Standard deviation. Special values are defined as:
  -9999.9  Missing value

Figure 1072: Data Format Structure for 3DPRX, G2, DFRNearSurface
hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

rainRate (Group in G1)
Conditional liquid water Rain Rate.
count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value
mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value
stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value
hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

snowRate (Group in G1)
Conditional Snowfall Rate.
count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value
mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value
stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value
hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

flagHeavyIcePrecip (Group in G1)
Counts of the occurrence of flagHeavyIcePrecip. Mean and std. dev. are set to missing.
The histogram contains counts of the integer flag values, with bins from 1 to 30.
count (4-byte integer, array size: ltL x lnL x chn x rt x st): Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st): mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st): Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin): Histogram. Special values are defined as:
-9999 Missing value

mixedPhRate (Group in G1)
Conditional Mixed Phase Precipitation Rate.

count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st): Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st): mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st): Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin): Histogram. Special values are defined as:
-9999 Missing value

precipRateESurface (Group in G1)
Conditional Estimated Surface Precipitation Rate.

count (4-byte integer, array size: ltL x lnL x chn x rt x st): Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st): mean. Special values are defined as:
-9999.9 Missing value
**stddev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipRateESurface2** (Group in G1)
Alternate Conditional Estimated Surface Precipitation Rate.

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stddev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipRateNearSurface** (Group in G1)
Conditional Precipitation Rate at Near Surface Level.

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stddev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value
rainRateNearSurface (Group in G1)
Unconditional liquid Rain Rate at Near Surface Level.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

snowRateNearSurface (Group in G1)
Conditional Snow Rate at Near Surface Level.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

mixedPhRateNearSurface (Group in G1)
Conditional Mixed Phase Precipitation Rate at Near Surface Level.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value
mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stddev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipWaterIntegrated (Group in G1)
Integrated Precipitable Water ($g/m^2$).

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stddev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipIceIntegrated (Group in G1)
Integrated Precipitable Ice

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stddev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value
hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
   -9999  Missing value

precipRateAve24 (Group in G1)
Average Precipitation Rate in 24hrs.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
   -9999  Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
   -9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
   -9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
   -9999  Missing value

zFactorCorrected (Group in G1)
Corrected Reflectivity

count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
   -9999  Missing value

mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
   -9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
   -9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
   -9999  Missing value

zFactorCorrectedESurface (Group in G1)
Corrected Reflectivity at the Estimated Surface
count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

zFactorCorrectedNearSurface (Group in G1)
Corrected Reflectivity at the Near Surface Level.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

zFactorMeasuredNearSurface (Group in G1)
Measured Reflectivity at the Near Surface Level.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value
**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**zFactorCorrectedDPR** (Group in G1)
Corrected Reflectivity from DPR

**count** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**zFactorCorrectedESurfaceDPR** (Group in G1)
Corrected Reflectivity from DPR at Estimated Surface.

**count** (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value
**zFactorCorrectedNearSurfaceDPR** (Group in G1)
Corrected Reflectivity from DPR at the Near Surface Level.

**count** (4-byte integer, array size: ltL x lnL x inst x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**zFactorMeasured** (Group in G1)
Measured Reflectivity

**count** (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**dm** (Group in G1)

**count** (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value
mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
  -9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
  -9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
  -9999  Missing value

dBNw (Group in G1)

count (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st):
Count. Special values are defined as:
  -9999  Missing value

mean (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
mean. Special values are defined as:
  -9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x chn x hgt x rt x st):
Standard deviation. Special values are defined as:
  -9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x chn x hgt x rt x st x bin):
Histogram. Special values are defined as:
  -9999  Missing value

epsilonDPR (Group in G1)

count (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st):
Count. Special values are defined as:
  -9999  Missing value

mean (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
mean. Special values are defined as:
  -9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x inst x hgt x rt x st):
Standard deviation. Special values are defined as:
  -9999.9  Missing value
**hist** (4-byte integer, array size: ltL x lnL x inst x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**epsilon** (Group in G1)

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**zeta** (Group in G1)
Integral of $0.2\ln(10)\alpha Z_m^\beta$ over the slant range path where $\alpha$ and $Z_m$ are functions of range.

**count** (4-byte integer, array size: ltL x lnL x chn x ang x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x ang x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x ang x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x ang x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**piaHB** (Group in G1)
Hitchfield-Bordan Path Integrated Attenuation for the slant range path.
**count** (4-byte integer, array size: \(ltL \times lnL \times chn \times ang \times rt \times st\)):
Count. Special values are defined as:
- \(-9999\) Missing value

**mean** (4-byte float, array size: \(ltL \times lnL \times chn \times ang \times rt \times st\)):
mean. Special values are defined as:
- \(-9999.9\) Missing value

**stdev** (4-byte float, array size: \(ltL \times lnL \times chn \times ang \times rt \times st\)):
Standard deviation. Special values are defined as:
- \(-9999.9\) Missing value

**hist** (4-byte integer, array size: \(ltL \times lnL \times chn \times ang \times rt \times st \times bin\)):
Histogram. Special values are defined as:
- \(-9999\) Missing value

**piaHybrid** (Group in G1)
Weighted Hybrid PIA between the HB solution and the SRT PIA.

**count** (4-byte integer, array size: \(ltL \times lnL \times chn \times ang \times rt \times st\)):
Count. Special values are defined as:
- \(-9999\) Missing value

**mean** (4-byte float, array size: \(ltL \times lnL \times chn \times ang \times rt \times st\)):
mean. Special values are defined as:
- \(-9999.9\) Missing value

**stdev** (4-byte float, array size: \(ltL \times lnL \times chn \times ang \times rt \times st\)):
Standard deviation. Special values are defined as:
- \(-9999.9\) Missing value

**hist** (4-byte integer, array size: \(ltL \times lnL \times chn \times ang \times rt \times st \times bin\)):
Histogram. Special values are defined as:
- \(-9999\) Missing value

**piaHybridDPR** (Group in G1)
Weighted Hybrid PIA between the HB solution and the SRT PIA for DPR.

**count** (4-byte integer, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\)):
Count. Special values are defined as:
- \(-9999\) Missing value

**mean** (4-byte float, array size: \(ltL \times lnL \times inst \times ang \times rt \times st\)):
mean. Special values are defined as:
- \(-9999.9\) Missing value
**STANDARD GPM PRODUCTS**

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
- 9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
- 9999  Missing value

**piaSRT** (Group in G1)
Path Integrated Attenuation from SRT.

**count** (4-byte integer, array size: ltL x lnL x chn x ang x rt x st):
Count. Special values are defined as:
- 9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x ang x rt x st):
mean. Special values are defined as:
- 9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x ang x rt x st):
Standard deviation. Special values are defined as:
- 9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x ang x rt x st x bin):
Histogram. Special values are defined as:
- 9999  Missing value

**piaSRTdpr** (Group in G1)
Path Integrated Attenuation from SRT DPR

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
- 9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
- 9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
- 9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
- 9999  Missing value
**piaFinal** (Group in G1)
Final Path Integrated Attenuation

**count** (4-byte integer, array size: ltL x lnL x chn x ang x rt x st):
Count. Special values are defined as:
   -9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x ang x rt x st):
mean. Special values are defined as:
   -9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x ang x rt x st):
Standard deviation. Special values are defined as:
   -9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x ang x rt x st x bin):
Histogram. Special values are defined as:
   -9999 Missing value

---

**piaFinalDPR** (Group in G1)
Final Path Integrated Attenuation from DPR

**count** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
   -9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
   -9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
   -9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
   -9999 Missing value

---

**piaFinalSubset** (Group in G1)
Final Path Integrated Attenuation Subset

**count** (4-byte integer, array size: ltL x lnL x chn x ang x rt x st):
Count. Special values are defined as:
   -9999 Missing value
mean (4-byte float, array size: ltL x lnL x chn x ang x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x chn x ang x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x chn x ang x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**piaFinalDPRsubset** (Group in G1)
Final Path Integrated Attenuation from DPR Subset

count (4-byte integer, array size: ltL x lnL x inst x ang x rt x st):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x inst x ang x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

hist (4-byte integer, array size: ltL x lnL x inst x ang x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**heightBB** (Group in G1)
Height of Bright Band.

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value
**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**heightBBnadir** (Group in G1)
Height of Bright Band from Nadir.

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**BBwidthNadir** (Group in G1)
Width of Bright Band at Nadir

**count** (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9  Missing value

**hist** (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999  Missing value

**heightStormTop** (Group in G1)
Storm Top Height
count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

BBwidth (Group in G1)
Bright Band Width

count (4-byte integer, array size: ltL x lnL x chn x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x chn x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x chn x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x chn x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

observationCounts (Group in G1)
Observation Counts

total (4-byte integer, array size: ltL x lnL x chn x st):
Total obs. Special values are defined as:
-9999 Missing value

localTime (4-byte integer, array size: ltL x lnL x chn x tim x st):
obs time. Special values are defined as:
-9999 Missing value
**pia** (4-byte integer, array size: ltL x lnL x chn x ang x st):  
obs PIA. Special values are defined as:  
  -9999 Missing value

**shallowRain** (4-byte integer, array size: ltL x lnL x chn x st):  
obs time. Special values are defined as:  
  -9999 Missing value

**precipRateLocalTime** (Group in G1)  
Precipitation Rate by Local Time

**count** (4-byte integer, array size: ltL x lnL x chn x tim x st):  
Count. Special values are defined as:  
  -9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x chn x tim x st):  
mean. Special values are defined as:  
  -9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x chn x tim x st):  
Standard deviation. Special values are defined as:  
  -9999.9 Missing value

**DFRmNearSurface** (Group in G1)  
DFRm at the Near Surface level

**count** (4-byte integer, array size: ltL x lnL x rt x st):  
Count. Special values are defined as:  
  -9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x rt x st):  
mean. Special values are defined as:  
  -9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x rt x st):  
Standard deviation. Special values are defined as:  
  -9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x rt x st x bin):  
Histogram. Special values are defined as:  
  -9999 Missing value

**DFRNearSurface** (Group in G1)  
DFR at the Near Surface level
count (4-byte integer, array size: ltL x lnL x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x rt x st):
Standard deviation. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipRateNearSurfaceUnconditional (4-byte float, array size: ltL x lnL x chn):
Rain, not conditioned on rain. Special values are defined as:
-9999.9 Missing value

precipProbabilityNearSurface (4-byte float, array size: ltL x lnL x chn):
Probability of rain. Special values are defined as:
-9999.9 Missing value

G2 (Grid)

G2_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipRate (Group in G2)
Conditional Precipitation Rate

count (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
rainRate (Group in G2)  
Conditional Liquid Rain Rate

**count** (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

snowRate (Group in G2)  
Conditional Snow Rate

**count** (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

flagHeavyIcePrecip (Group in G2)  
Counts of the occurrence of flagHeavyIcePrecip. Mean and std. dev. are set to missing.
The histogram contains counts of the integer flag values, with bins from 1 to 30.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value
mixedPhRate (Group in G2)
Conditional Precipitation Rate of Mixed Phase

- **count** (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
  Count. Special values are defined as:
  - -9999 Missing value

- **mean** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
  mean. Special values are defined as:
  - -9999.9 Missing value

- **stdev** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
  Standard deviation. Special values are defined as:
  - -9999.9 Missing value

precipRateESurface (Group in G2)
Conditional Estimated Precipitation Rate at the Surface

- **count** (4-byte integer, array size: ltH x lnH x chn x rt):
  Count. Special values are defined as:
  - 9999 Missing value

- **mean** (4-byte float, array size: ltH x lnH x chn x rt):
  mean. Special values are defined as:
  - -9999.9 Missing value

- **stdev** (4-byte float, array size: ltH x lnH x chn x rt):
  Standard deviation. Special values are defined as:
  - -9999.9 Missing value

precipRateESurface2 (Group in G2)
Alternate Conditional Estimated Precipitation Rate at the Surface

- **count** (4-byte integer, array size: ltH x lnH x chn x rt):
  Count. Special values are defined as:
  - 9999 Missing value

- **mean** (4-byte float, array size: ltH x lnH x chn x rt):
  mean. Special values are defined as:
  - -9999.9 Missing value

- **stdev** (4-byte float, array size: ltH x lnH x chn x rt):
  Standard deviation. Special values are defined as:
  - -9999.9 Missing value
**precipRateNearSurface** (Group in G2)
Conditional Precipitation Rate at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**rainRateNearSurface** (Group in G2)
Conditional Liquid Rain Rate at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**snowRateNearSurface** (Group in G2)
Conditional Snow Rate at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**mixedPhRateNearSurface** (Group in G2)
Conditional Precipitation Rate of Mixed Phase at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

---

**precipWaterIntegrated** (Group in G2)
Integrated Precipitable Water \((g/m^2)\).

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

---

**precipIceIntegrated** (Group in G2)
Integrated Precipitable Ice

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**precipRateAve24** (Group in G2)
Conditional Precipitation Rate Averaged for 24hrs.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**zFactorCorrected** (Group in G2)
Corrected Reflectivity.

**count** (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**zFactorCorrectedESurface** (Group in G2)
Corrected Reflectivity Estimate at the Surface

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**zFactorCorrectedNearSurface** (Group in G2)
Corrected Reflectivity at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:

-9999 = Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 = Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 = Missing value

---

**zFactorMeasuredNearSurface** (Group in G2)
Measured Reflectivity at the Near Surface Level.

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999 = Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9 = Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9 = Missing value

---

**zFactorCorrectedDPR** (Group in G2)
Corrected Reflectivity from DPR

**count** (4-byte integer, array size: ltH x lnH x inst x hgt x rt):
Count. Special values are defined as:
-9999 = Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
mean. Special values are defined as:
-9999.9 = Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9 = Missing value
**zFactorCorrectedESurfaceDPR** (Group in G2)
Estimated Corrected Reflectivity at the Surface

**count** (4-byte integer, array size: ltH x lnH x inst x rt):
Count. Special values are defined as:
- -9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x rt):
mean. Special values are defined as:
- -9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x rt):
Standard deviation. Special values are defined as:
- -9999.9 Missing value

**zFactorCorrectedNearSurfaceDPR** (Group in G2)
Corrected Reflectivity at the Near Surface Level for DPR

**count** (4-byte integer, array size: ltH x lnH x inst x rt):
Count. Special values are defined as:
- -9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x rt):
mean. Special values are defined as:
- -9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x rt):
Standard deviation. Special values are defined as:
- -9999.9 Missing value

**zFactorMeasured** (Group in G2)
Corrected Reflectivity

**count** (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
- -9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
- -9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
- -9999.9 Missing value
dm (Group in G2)
Mean Mass-Weighted Drop Diameter

**count** (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

dBNw (Group in G2)
Normalized Drop Concentration Parameter

**count** (4-byte integer, array size: ltH x lnH x chn x hgt x rt):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

epsilonDPR (Group in G2)

**count** (4-byte integer, array size: ltH x lnH x inst x hgt x rt):
Count. Special values are defined as:
-9999  Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
mean. Special values are defined as:
-9999.9  Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x hgt x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value
**epsilon (Group in G2)**

**count** (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
  -9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
  -9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
  -9999.9 Missing value

**zeta (Group in G2)**
Integral of $0.2*\ln(10)\alpha Zm^\beta$ over the slant range path where $\alpha$ and $Zm$ are functions of range.

**count** (4-byte integer, array size: ltH x lnH x chn x ang x rt):
Count. Special values are defined as:
  -9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x ang x rt):
mean. Special values are defined as:
  -9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x ang x rt):
Standard deviation. Special values are defined as:
  -9999.9 Missing value

**piaHB (Group in G2)**
Hitchfield-Bordan Path Integrated Attenuation for the slant range path.

**count** (4-byte integer, array size: ltH x lnH x chn x ang x rt):
Count. Special values are defined as:
  -9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x ang x rt):
mean. Special values are defined as:
  -9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x ang x rt):
Standard deviation. Special values are defined as:
  -9999.9 Missing value
**piaHybrid** (Group in G2)
Weighted Hybrid PIA between the HB solution and the SRT PIA.

**count** (4-byte integer, array size: ltH x lnH x chn x ang x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x ang x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x ang x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**piaHybridDPR** (Group in G2)
Weighted Hybrid PIA between the HB solution and the SRT PIA for DPR.

**count** (4-byte integer, array size: ltH x lnH x inst x ang x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x inst x ang x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**piaSRT** (Group in G2)
Path Integrated Attenuation from SRT.

**count** (4-byte integer, array size: ltH x lnH x chn x ang x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x chn x ang x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x chn x ang x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
**piaSRTdpr** (Group in G2)
Path Integrated Attenuation from SRT for DPR.

- **count** (4-byte integer, array size: ltH x lnH x inst x ang x rt):
  Count. Special values are defined as:
  -9999   Missing value

- **mean** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  mean. Special values are defined as:
  -9999.9   Missing value

- **stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  Standard deviation. Special values are defined as:
  -9999.9   Missing value

**piaFinal** (Group in G2)
Final Path Integrated Attenuation Estimate.

- **count** (4-byte integer, array size: ltH x lnH x chn x ang x rt):
  Count. Special values are defined as:
  -9999   Missing value

- **mean** (4-byte float, array size: ltH x lnH x chn x ang x rt):
  mean. Special values are defined as:
  -9999.9   Missing value

- **stdev** (4-byte float, array size: ltH x lnH x chn x ang x rt):
  Standard deviation. Special values are defined as:
  -9999.9   Missing value

**piaFinalDPR** (Group in G2)
Final Path Integrated Attenuation Estimate for DPR.

- **count** (4-byte integer, array size: ltH x lnH x inst x ang x rt):
  Count. Special values are defined as:
  -9999   Missing value

- **mean** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  mean. Special values are defined as:
  -9999.9   Missing value

- **stdev** (4-byte float, array size: ltH x lnH x inst x ang x rt):
  Standard deviation. Special values are defined as:
  -9999.9   Missing value
heightBB (Group in G2)
Height Of the Bright Band.

count (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

heightStormTop (Group in G2)
Height of the Storm Top.

count (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value

BBwidth (Group in G2)
Bright Band Width

count (4-byte integer, array size: ltH x lnH x chn x rt):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: ltH x lnH x chn x rt):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: ltH x lnH x chn x rt):
Standard deviation. Special values are defined as:
-9999.9  Missing value
**observationCounts** (Group in G2)
Observation Counts.

**total** (4-byte integer, array size: ltH x lnH x chn):
Total obs. Special values are defined as:
-9999 Missing value

**pia** (4-byte integer, array size: ltH x lnH x chn x ang):
obs PIA. Special values are defined as:
-9999 Missing value

**shallowRain** (4-byte integer, array size: ltH x lnH x chn):
obs time. Special values are defined as:
-9999 Missing value

**DFRmNearSurface** (Group in G2)
DFRm at the Near Surface level

**count** (4-byte integer, array size: ltH x lnH x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value

**DFRNearSurface** (Group in G2)
DFR at the Near Surface level

**count** (4-byte integer, array size: ltH x lnH x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x rt):
Standard deviation. Special values are defined as:
-9999.9 Missing value
precipRateNearSurfaceUnconditional (4-byte float, array size: ltH x lnH x chn):
Rain, not conditioned on rain. Special values are defined as:
-9999.9 Missing value

precipProbabilityNearSurface (4-byte float, array size: ltH x lnH x chn):
Probability of rain. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_3DPRX_H_
#define _TK_3DPRX_H_

#ifndef _L3DPRX_G2_DFRNEARSURFACE_
#define _L3DPRX_G2_DFRNEARSURFACE_

typedef struct {
    int count[3][1440][536];
    float mean[3][1440][536];
    float stdev[3][1440][536];
} L3DPRX_G2_DFRNEARSURFACE;

#endif

#ifndef _L3DPRX_G2_DFRMNEARSURFACE_
#define _L3DPRX_G2_DFRMNEARSURFACE_

typedef struct {
    int count[3][1440][536];
    float mean[3][1440][536];
    float stdev[3][1440][536];
} L3DPRX_G2_DFRMNEARSURFACE;

#endif

#ifndef _L3DPRX_G2_OBSERVATIONCOUNTS_
#define _L3DPRX_G2_OBSERVATIONCOUNTS_

typedef struct {
    int total[7][1440][536];
    int pia[7][7][1440][536];
    int shallowRain[7][1440][536];
} L3DPRX_G2_OBSERVATIONCOUNTS;

#endif
```
typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_BBWIDTH;

#endif

#endif

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_HEIGHTSTORMTOP;

#endif

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_HEIGHTBB;

#endif

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPRX_G2_PIAFINALDPR;
#ifndef _L3DPRX_G2_PIAFINAL_
#define _L3DPRX_G2_PIAFINAL_

typedef struct {
    int count[3][7][7][1440][536];
    float mean[3][7][7][1440][536];
    float stdev[3][7][7][1440][536];
} L3DPRX_G2_PIAFINAL;

#endif

#ifndef _L3DPRX_G2_PIASRTDPR_
#define _L3DPRX_G2_PIASRTDPR_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPRX_G2_PIASRTDPR;

#endif

#ifndef _L3DPRX_G2_PIASRT_
#define _L3DPRX_G2_PIASRT_

typedef struct {
    int count[3][7][7][1440][536];
    float mean[3][7][7][1440][536];
    float stdev[3][7][7][1440][536];
} L3DPRX_G2_PIASRT;

#endif

#ifndef _L3DPRX_G2_PIAHYBRIDDPR_
#define _L3DPRX_G2_PIAHYBRIDDPR_

typedef struct {
    int count[3][7][4][1440][536];
    float mean[3][7][4][1440][536];
    float stdev[3][7][4][1440][536];
} L3DPRX_G2_PIAHYBRIDDPR;
#ifndef _L3DPRX_G2_PIAHYBRID_
#define _L3DPRX_G2_PIAHYBRID_

typedef struct {
    int count[3][7][7][1440][536];
    float mean[3][7][7][1440][536];
    float stdev[3][7][7][1440][536];
} L3DPRX_G2_PIAHYBRID;
#endif

#endif

#ifndef _L3DPRX_G2_PIAHB_
#define _L3DPRX_G2_PIAHB_

typedef struct {
    int count[3][7][7][1440][536];
    float mean[3][7][7][1440][536];
    float stdev[3][7][7][1440][536];
} L3DPRX_G2_PIAHB;
#endif

#endif

#ifndef _L3DPRX_G2_ZETA_
#define _L3DPRX_G2_ZETA_

typedef struct {
    int count[3][7][7][1440][536];
    float mean[3][7][7][1440][536];
    float stdev[3][7][7][1440][536];
} L3DPRX_G2_ZETA;
#endif

#endif

#ifndef _L3DPRX_G2_EPSILON_
#define _L3DPRX_G2_EPSILON_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
} L3DPRX_G2_EPSILON;
typedef struct {
    int count[3][5][4][1440][536];
    float mean[3][5][4][1440][536];
    float stdev[3][5][4][1440][536];
} L3DPRX_G2_EPSILON;

#endif

#ifndef _L3DPRX_G2_EPSILON_DPR_
#define _L3DPRX_G2_EPSILON_DPR_

typedef struct {
    int count[3][5][7][1440][536];
    float mean[3][5][7][1440][536];
    float stdev[3][5][7][1440][536];
} L3DPRX_G2_EPSILON_DPR;

#endif

#ifndef _L3DPRX_G2_DBNW_
#define _L3DPRX_G2_DBNW_

typedef struct {
    int count[3][5][7][1440][536];
    float mean[3][5][7][1440][536];
    float stdev[3][5][7][1440][536];
} L3DPRX_G2_DBNW;

#endif

#ifndef _L3DPRX_G2_DM_
#define _L3DPRX_G2_DM_

typedef struct {
    int count[3][5][7][1440][536];
    float mean[3][5][7][1440][536];
    float stdev[3][5][7][1440][536];
} L3DPRX_G2_DM;

#endif

#ifndef _L3DPRX_G2_ZFACTORMEASURED_
#define _L3DPRX_G2_ZFACTORMEASURED_

typedef struct {
    int count[3][5][7][1440][536];

```c
float mean[3][5][7][1440][536];
float stdev[3][5][7][1440][536];
} L3DPRX_G2_ZFACTORMEASURED;
#endif

#ifndef _L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACEDPR_
#define _L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACEDPR_
typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACEDPR;
#endif

#ifndef _L3DPRX_G2_ZFACTORCORRECTEDSURFACEDPR_
#define _L3DPRX_G2_ZFACTORCORRECTEDSURFACEDPR_
typedef struct {
    int count[3][4][1440][536];
    float mean[3][4][1440][536];
    float stdev[3][4][1440][536];
} L3DPRX_G2_ZFACTORCORRECTEDSURFACEDPR;
#endif

#ifndef _L3DPRX_G2_ZFACTORCORRECTEDDPR_
#define _L3DPRX_G2_ZFACTORCORRECTEDDPR_
typedef struct {
    int count[3][5][4][1440][536];
    float mean[3][5][4][1440][536];
    float stdev[3][5][4][1440][536];
} L3DPRX_G2_ZFACTORCORRECTEDDPR;
#endif

#ifndef _L3DPRX_G2_ZFACTORMEASUREDNEARSURFACE_
#define _L3DPRX_G2_ZFACTORMEASUREDNEARSURFACE_
typedef struct {
```
typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_ZFACTORMEASUREDNEARSURFACE;

#endif

#ifdef _L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACE_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACE;

#endif

#ifdef _L3DPRX_G2_ZFACTORCORRECTEDESURFACE_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_ZFACTORCORRECTEDESURFACE;

#endif

#ifdef _L3DPRX_G2_ZFACTORCORRECTED_

typedef struct {
    int count[3][5][7][1440][536];
    float mean[3][5][7][1440][536];
    float stdev[3][5][7][1440][536];
} L3DPRX_G2_ZFACTORCORRECTED;

#endif

#ifndef _L3DPRX_G2_PRECIPRATEAVE24_
#define _L3DPRX_G2_PRECIPRATEAVE24_

#endif
typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_PRECIPRATEAVE24;

#endif

#ifndef _L3DPRX_G2_PRECIPICEINTEGRATED_
#define _L3DPRX_G2_PRECIPICEINTEGRATED_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_PRECIPICEINTEGRATED;

#endif

#ifndef _L3DPRX_G2_PRECIPWATERINTEGRATED_
#define _L3DPRX_G2_PRECIPWATERINTEGRATED_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_PRECIPWATERINTEGRATED;

#endif

#ifndef _L3DPRX_G2_MIXEDPHRATENEARSURFACE_
#define _L3DPRX_G2_MIXEDPHRATENEARSURFACE_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_MIXEDPHRATENEARSURFACE;

#endif

#ifndef _L3DPRX_G2_SNOWRATENEARSURFACE_
#define _L3DPRX_G2_SNOWRATENEARSURFACE_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_SNOWRATENEARSURFACE;

#endif
typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_SNOWRATENEARSURFACE;

#endif

#ifndef _L3DPRX_G2_RAINRATENEARSURFACE_
#define _L3DPRX_G2_RAINRATENEARSURFACE_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_RAINRATENEARSURFACE;

#endif

#ifndef _L3DPRX_G2_PRECIPRATENEARSURFACE_
#define _L3DPRX_G2_PRECIPRATENEARSURFACE_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_PRECIPRATENEARSURFACE;

#endif

#ifndef _L3DPRX_G2_PRECIPRATEESURFACE2_
#define _L3DPRX_G2_PRECIPRATEESURFACE2_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_PRECIPRATEESURFACE2;

#endif
#define _L3DPRX_G2_PRECIPRATEESURFACE_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_PRECIPRATEESURFACE;
#endif

#ifndef _L3DPRX_G2_MIXEDPHRATE_
#define _L3DPRX_G2_MIXEDPHRATE_

typedef struct {
    int count[3][5][7][1440][536];
    float mean[3][5][7][1440][536];
    float stdev[3][5][7][1440][536];
} L3DPRX_G2_MIXEDPHRATE;
#endif

#ifndef _L3DPRX_G2_FLAGHEAVYICEPRECIP_
#define _L3DPRX_G2_FLAGHEAVYICEPRECIP_

typedef struct {
    int count[3][7][1440][536];
    float mean[3][7][1440][536];
    float stdev[3][7][1440][536];
} L3DPRX_G2_FLAGHEAVYICEPRECIP;
#endif

#ifndef _L3DPRX_G2_SNOWRATE_
#define _L3DPRX_G2_SNOWRATE_

typedef struct {
    int count[3][5][7][1440][536];
    float mean[3][5][7][1440][536];
    float stdev[3][5][7][1440][536];
} L3DPRX_G2_SNOWRATE;
#endif
#ifndef _L3DPRX_G2_RAINRATE_
define _L3DPRX_G2_RAINRATE_

typedef struct {
   int count[3][5][7][1440][536];
   float mean[3][5][7][1440][536];
   float stdev[3][5][7][1440][536];
} L3DPRX_G2_RAINRATE;
#endif

#ifndef _L3DPRX_G2_PRECIPRATE_
define _L3DPRX_G2_PRECIPRATE_

typedef struct {
   int count[3][5][7][1440][536];
   float mean[3][5][7][1440][536];
   float stdev[3][5][7][1440][536];
} L3DPRX_G2_PRECIPRATE;
#endif

#ifndef _L3DPRX_G2_
define _L3DPRX_G2_

typedef struct {
   L3DPRX_G2_PRECIPRATE precipRate;
   L3DPRX_G2_RAINRATE rainRate;
   L3DPRX_G2_SNOWRATE snowRate;
   L3DPRX_G2_FLAGHEAVYICEPREcip flagHeavyIcePrecip;
   L3DPRX_G2_MIXEDPHRATE mixedPhRate;
   L3DPRX_G2_PRECIPRATEESURFACE precipRateESurface;
   L3DPRX_G2_PRECIPRATEESURFACE2 precipRateESurface2;
   L3DPRX_G2_PRECIPATENEARSURFACE precipRateNearSurface;
   L3DPRX_G2_RAINRATENEARSURFACE rainRateNearSurface;
   L3DPRX_G2_SNOWRATENEARSURFACE snowRateNearSurface;
   L3DPRX_G2_MIXEDPHRATENEARSURFACE mixedPhRateNearSurface;
   L3DPRX_G2_PRECIPWATERINTEGRATED precipWaterIntegrated;
   L3DPRX_G2_PRECIPICEINTEGRATED precipIceIntegrated;
   L3DPRX_G2_PRECIPRATEAVE24 precipRateAve24;
   L3DPRX_G2_ZFACTORCORRECTED zFactorCorrected;
   L3DPRX_G2_ZFACTORCORRECTEDESURFACE zFactorCorrectedESurface;
   L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACE zFactorCorrectedNearSurface;
} L3DPRX_G2;
L3DPRX_G2_ZFACTORMEASUREDNEARSURFACE zFactorMeasuredNearSurface;
L3DPRX_G2_ZFACTORCORRECTEDDPR zFactorCorrectedDPR;
L3DPRX_G2_ZFACTORCORRECTEDESURFACEDPR zFactorCorrectedESurfaceDPR;
L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACEDPR zFactorCorrectedNearSurfaceDPR;
L3DPRX_G2_ZFACTORMEASURED zFactorMeasured;
L3DPRX_G2_DM dm;
L3DPRX_G2_DBNW dBNw;
L3DPRX_G2_EPSILONDPR epsilonDPR;
L3DPRX_G2_EPSILON epsilon;
L3DPRX_G2_ZETA zeta;
L3DPRX_G2_PIAHB piaHB;
L3DPRX_G2_PIAHYBRID piaHybrid;
L3DPRX_G2_PIAHYBRIDDPR piaHybridDPR;
L3DPRX_G2_PIASRT piaSRT;
L3DPRX_G2_PIASRTDPR piaSRTdpr;
L3DPRX_G2_PIAFINAL piaFinal;
L3DPRX_G2_PIAFINALDPR piaFinalDPR;
L3DPRX_G2_HEIGHTBB heightBB;
L3DPRX_G2_HEIGHTSTORMTOP heightStormTop;
L3DPRX_G2_BBWIDTH BBwidth;
L3DPRX_G2_OBSERVATIONCOUNTS observationCounts;
L3DPRX_G2_DFRMNEARSURFACE DFRmNearSurface;
L3DPRX_G2_DFRNEARSURFACE DFRNearSurface;
float precipRateNearSurfaceUnconditional[7][1440][536];
float precipProbabilityNearSurface[7][1440][536];
} L3DPRX_G2;

#endif

ifndef _L3DPRX_G1_DFRNEARSURFACE_
define _L3DPRX_G1_DFRNEARSURFACE_

typedef struct {
    int count[3][3][72][28];
    float mean[3][3][72][28];
    float stdev[3][3][72][28];
    int hist[30][3][3][72][28];
} L3DPRX_G1_DFRNEARSURFACE;
#endif

ifndef _L3DPRX_G1_DFRMNEARSURFACE_
define _L3DPRX_G1_DFRMNEARSURFACE_


typedef struct {
    int count[3][3][72][28];
    float mean[3][3][72][28];
    float stdev[3][3][72][28];
    int hist[30][3][3][72][28];
} L3DPRX_G1_DFRMNEARSURFACE;

#endif

#ifndef _L3DPRX_G1_PRECIPRATELOCALTIME_
#define _L3DPRX_G1_PRECIPRATELOCALTIME_

typedef struct {
    int count[3][24][7][72][28];
    float mean[3][24][7][72][28];
    float stdev[3][24][7][72][28];
} L3DPRX_G1_PRECIPRATELOCALTIME;
#endif

#ifndef _L3DPRX_G1_OBSERVATIONCOUNTS_
#define _L3DPRX_G1_OBSERVATIONCOUNTS_

typedef struct {
    int total[3][7][72][28];
    int localTime[3][24][7][72][28];
    int pia[3][7][72][28];
    int shallowRain[3][7][72][28];
} L3DPRX_G1_OBSERVATIONCOUNTS;
#endif

#ifndef _L3DPRX_G1_BBWIDTH_
#define _L3DPRX_G1_BBWIDTH_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_BBWIDTH;
#ifndef _L3DPRX_G1_HEIGHTSTORMTOP_
#define _L3DPRX_G1_HEIGHTSTORMTOP_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_HEIGHTSTORMTOP;
#endif

#ifndef _L3DPRX_G1_BBWIDTHNADIR_
#define _L3DPRX_G1_BBWIDTHNADIR_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_BBWIDTHNADIR;
#endif

#ifndef _L3DPRX_G1_HEIGHTBBNADIR_
#define _L3DPRX_G1_HEIGHTBBNADIR_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_HEIGHTBBNADIR;
#endif

#ifndef _L3DPRX_G1_HEIGHTBB_
#define _L3DPRX_G1_HEIGHTBB_

typedef struct {
    int count[3][3][7][72][28];
} L3DPRX_G1_HEIGHTBB;
#endif
float mean[3][3][7][72][28];
float stdev[3][3][7][72][28];
int hist[30][3][3][7][72][28];
} L3DPRX_G1_HEIGHTBB;

#endif

#define _L3DPRX_G1_PIAFINALDPRSUBSET_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPRX_G1_PIAFINALDPRSUBSET;

#endif

#define _L3DPRX_G1_PIAFINALSUBSET_

typedef struct {
    int count[3][3][7][7][72][28];
    float mean[3][3][7][7][72][28];
    float stdev[3][3][7][7][72][28];
    int hist[30][3][3][7][7][72][28];
} L3DPRX_G1_PIAFINALSUBSET;

#endif

#define _L3DPRX_G1_PIAFINALDPR_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPRX_G1_PIAFINALDPR;

#endif
#ifndef _L3DPRX_G1_PIAFINAL_
#define _L3DPRX_G1_PIAFINAL_

typedef struct {
    int count[3][3][7][7][72][28];
    float mean[3][3][7][7][72][28];
    float stdev[3][3][7][7][72][28];
    int hist[30][3][3][7][7][72][28];
} L3DPRX_G1_PIAFINAL;

#endif

#ifndef _L3DPRX_G1 PIASRTDPR_
#define _L3DPRX_G1 PIASRTDPR_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
    int hist[30][3][3][7][4][72][28];
} L3DPRX_G1 PIASRTDPR;

#endif

#ifndef _L3DPRX_G1_PIASRT_
#define _L3DPRX_G1_PIASRT_

typedef struct {
    int count[3][3][7][7][72][28];
    float mean[3][3][7][7][72][28];
    float stdev[3][3][7][7][72][28];
    int hist[30][3][3][7][7][72][28];
} L3DPRX_G1 PIASRT;

#endif

#ifndef _L3DPRX_G1_PIHYBRIDDPR_
#define _L3DPRX_G1_PIHYBRIDDPR_

typedef struct {
    int count[3][3][7][4][72][28];
    float mean[3][3][7][4][72][28];
    float stdev[3][3][7][4][72][28];
} L3DPRX_G1 PIHYBRIDDPR;

#endif
int hist[30][3][3][7][7][72][28];
} L3DPRX_G1_PIAHYBRIDDPR;

#endif

#ifndef _L3DPRX_G1_PIAHYBRID_
#define _L3DPRX_G1_PIAHYBRID_

typedef struct {
    int count[3][3][7][7][72][28];
    float mean[3][3][7][7][72][28];
    float stdev[3][3][7][7][72][28];
    int hist[30][3][3][7][7][72][28];
} L3DPRX_G1_PIAHYBRID;

#endif

#ifndef _L3DPRX_G1_PIAHB_
#define _L3DPRX_G1_PIAHB_

typedef struct {
    int count[3][3][7][7][72][28];
    float mean[3][3][7][7][72][28];
    float stdev[3][3][7][7][72][28];
    int hist[30][3][3][7][7][72][28];
} L3DPRX_G1_PIAHB;

#endif

#ifndef _L3DPRX_G1_ZETA_
#define _L3DPRX_G1_ZETA_

typedef struct {
    int count[3][3][7][7][72][28];
    float mean[3][3][7][7][72][28];
    float stdev[3][3][7][7][72][28];
    int hist[30][3][3][7][7][72][28];
} L3DPRX_G1_ZETA;

#endif

#ifndef _L3DPRX_G1_EPSILON_
#define _L3DPRX_G1_EPSILON_
typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_EPSILON;

#endif

#ifndef _L3DPRX_G1_EPSILON_DPR_
#define _L3DPRX_G1_EPSILON_DPR_

typedef struct {
    int count[3][3][5][4][72][28];
    float mean[3][3][5][4][72][28];
    float stdev[3][3][5][4][72][28];
    int hist[30][3][3][5][4][72][28];
} L3DPRX_G1_EPSILON_DPR;

#endif

#ifndef _L3DPRX_G1_DBNW_
#define _L3DPRX_G1_DBNW_

typedef struct {
    int count[3][3][5][7][72][28];
    float mean[3][3][5][7][72][28];
    float stdev[3][3][5][7][72][28];
    int hist[30][3][3][5][7][72][28];
} L3DPRX_G1_DBNW;

#endif

#ifndef _L3DPRX_G1_DM_
#define _L3DPRX_G1_DM_

typedef struct {
    int count[3][3][5][7][72][28];
    float mean[3][3][5][7][72][28];
    float stdev[3][3][5][7][72][28];
    int hist[30][3][3][5][7][72][28];
} L3DPRX_G1_DM;

}
#endif

#ifndef _L3DPRX_G1_ZFACTORMEASURED_
define _L3DPRX_G1_ZFACTORMEASURED_

typedef struct {
  int count[3][3][5][7][72][28];
  float mean[3][3][5][7][72][28];
  float stdev[3][3][5][7][72][28];
  int hist[30][3][3][5][7][72][28];
} L3DPRX_G1_ZFACTORMEASURED;
define

#endif

#ifndef _L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACEDPR_
define _L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACEDPR_

typedef struct {
  int count[3][3][4][72][28];
  float mean[3][3][4][72][28];
  float stdev[3][3][4][72][28];
  int hist[30][3][3][4][72][28];
} L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACEDPR;
define

#endif

#ifndef _L3DPRX_G1_ZFACTORCORRECTEDDESURFACEDPR_
define _L3DPRX_G1_ZFACTORCORRECTEDDESURFACEDPR_

typedef struct {
  int count[3][3][4][72][28];
  float mean[3][3][4][72][28];
  float stdev[3][3][4][72][28];
  int hist[30][3][3][4][72][28];
} L3DPRX_G1_ZFACTORCORRECTEDDESURFACEDPR;
define

#endif

#ifndef _L3DPRX_G1_ZFACTORCORRECTEDDPR_
define _L3DPRX_G1_ZFACTORCORRECTEDDPR_

typedef struct {

} L3DPRX_G1_ZFACTORCORRECTEDDPR;
int count[3][3][5][4][72][28];
float mean[3][3][5][4][72][28];
float stdev[3][3][5][4][72][28];
int hist[30][3][3][5][4][72][28];
} L3DPRX_G1_ZFACTORCORRECTEDDPR;

#endif

#ifdef _L3DPRX_G1_ZFACTORMEASUREDNEARSURFACE_
#define _L3DPRX_G1_ZFACTORMEASUREDNEARSURFACE_

typedef struct {
int count[3][3][7][72][28];
float mean[3][3][7][72][28];
float stdev[3][3][7][72][28];
int hist[30][3][3][7][72][28];
} L3DPRX_G1_ZFACTORMEASUREDNEARSURFACE;
#endif

#ifdef _L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACE_
#define _L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACE_

typedef struct {
int count[3][3][7][72][28];
float mean[3][3][7][72][28];
float stdev[3][3][7][72][28];
int hist[30][3][3][7][72][28];
} L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACE;
#endif

#ifdef _L3DPRX_G1_ZFACTORCORRECTEDDESURFACE_
#define _L3DPRX_G1_ZFACTORCORRECTEDDESURFACE_

typedef struct {
int count[3][3][7][72][28];
float mean[3][3][7][72][28];
float stdev[3][3][7][72][28];
int hist[30][3][3][7][72][28];
} L3DPRX_G1_ZFACTORCORRECTEDDESURFACE;
#endif
#ifndef _L3DPRX_G1_ZFACTORCORRECTED_
define _L3DPRX_G1_ZFACTORCORRECTED_

typedef struct {
    int count[3][3][5][7][72][28];
    float mean[3][3][5][7][72][28];
    float stdev[3][3][5][7][72][28];
    int hist[30][3][3][5][7][72][28];
} L3DPRX_G1_ZFACTORCORRECTED;

#endif

#ifndef _L3DPRX_G1_PRECIPRATEAVE24_
define _L3DPRX_G1_PRECIPRATEAVE24_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_PRECIPRATEAVE24;

#endif

#ifndef _L3DPRX_G1_PRECIPICEINTEGRATED_
define _L3DPRX_G1_PRECIPICEINTEGRATED_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_PRECIPICEINTEGRATED;

#endif

#ifndef _L3DPRX_G1_PRECIPWATERINTEGRATED_
define _L3DPRX_G1_PRECIPWATERINTEGRATED_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
} L3DPRX_G1_PRECIPWATERINTEGRATED;

#endif
typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_MIXEDPHRATENEARSURFACE;

#endif

#ifndef _L3DPRX_G1_MIXEDPHRATENEARSURFACE_
#define _L3DPRX_G1_MIXEDPHRATENEARSURFACE_

#endif

#ifndef _L3DPRX_G1_SNOWRATENEARSURFACE_
#define _L3DPRX_G1_SNOWRATENEARSURFACE_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_SNOWRATENEARSURFACE;

#endif

#ifndef _L3DPRX_G1_RAINRATENEARSURFACE_
#define _L3DPRX_G1_RAINRATENEARSURFACE_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_RAINRATENEARSURFACE;

#endif

#ifndef _L3DPRX_G1_PRECIPRATENEARSURFACE_
#define _L3DPRX_G1_PRECIPRATENEARSURFACE_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_PRECIPRATENEARSURFACE;

#endif
#define _L3DPRX_G1_PRECIPRATENEARSURFACE_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_PRECIPRATENEARSURFACE;

#endif

#ifdef _L3DPRX_G1_PRECIPRATEESURFACE2_
#define _L3DPRX_G1_PRECIPRATEESURFACE2_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_PRECIPRATEESURFACE2;

#endif

#ifdef _L3DPRX_G1_PRECIPRATEESURFACE_
#define _L3DPRX_G1_PRECIPRATEESURFACE_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_PRECIPRATEESURFACE;

#endif

#ifdef _L3DPRX_G1_MIXEDPHRATE_
#define _L3DPRX_G1_MIXEDPHRATE_

typedef struct {
    int count[3][3][5][7][72][28];
    float mean[3][3][5][7][72][28];
    float stdev[3][3][5][7][72][28];
    int hist[30][3][3][5][7][72][28];
} L3DPRX_G1_MIXEDPHRATE;

#endif
} L3DPRX_G1_MIXEDPHRATE;
#endif

#ifndef _L3DPRX_G1_FLAGHEAVYICEPRECIP_
#define _L3DPRX_G1_FLAGHEAVYICEPRECIP_

typedef struct {
    int count[3][3][7][72][28];
    float mean[3][3][7][72][28];
    float stdev[3][3][7][72][28];
    int hist[30][3][3][7][72][28];
} L3DPRX_G1_FLAGHEAVYICEPRECIP;
#endif

#ifndef _L3DPRX_G1_SNOWRATE_
#define _L3DPRX_G1_SNOWRATE_

typedef struct {
    int count[3][3][5][7][72][28];
    float mean[3][3][5][7][72][28];
    float stdev[3][3][5][7][72][28];
    int hist[30][3][3][5][7][72][28];
} L3DPRX_G1_SNOWRATE;
#endif

#ifndef _L3DPRX_G1_RAINRATE_
#define _L3DPRX_G1_RAINRATE_

typedef struct {
    int count[3][3][5][7][72][28];
    float mean[3][3][5][7][72][28];
    float stdev[3][3][5][7][72][28];
    int hist[30][3][3][5][7][72][28];
} L3DPRX_G1_RAINRATE;
#endif

#ifndef _L3DPRX_G1_PRECIPRATE_
#define _L3DPRX_G1_PRECIPRATE_

typedef struct {
    int count[3][3][5][7][72][28];
    float mean[3][3][5][7][72][28];
    float stdev[3][3][5][7][72][28];
    int hist[30][3][3][5][7][72][28];
} L3DPRX_G1_PRECIPRATE;

#endif

#ifndef _L3DPRX_G1_
#define _L3DPRX_G1_

typedef struct {
    L3DPRX_G1_PRECIPRATE precipRate;
    L3DPRX_G1_RAINRATE rainRate;
    L3DPRX_G1_SNOWRATE snowRate;
    L3DPRX_G1_FLAGHEAVYICEPRECIP flagHeavyIcePrecip;
    L3DPRX_G1_MIXEDPHRATE mixedPhRate;
    L3DPRX_G1_PRECIPRATEESURFACE precipRateESurface;
    L3DPRX_G1_PRECIPRATEESURFACE2 precipRateESurface2;
    L3DPRX_G1_PRECIPRATENEARSURFACE precipRateNearSurface;
    L3DPRX_G1_RAINRATENEARSURFACE rainRateNearSurface;
    L3DPRX_G1_SNOWRATENEARSURFACE snowRateNearSurface;
    L3DPRX_G1_MIXEDPHRATENEARSURFACE mixedPhRateNearSurface;
    L3DPRX_G1_PRECIPWATERINTEGRATED precipWaterIntegrated;
    L3DPRX_G1_PRECIPICEINTEGRATED precipIceIntegrated;
    L3DPRX_G1_PRECIPRATENAVE24 precipRateAve24;
    L3DPRX_G1_ZFACTORCORRECTED zFactorCorrected;
    L3DPRX_G1_ZFACTORCORRECTEDESURFACE zFactorCorrectedESurface;
    L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACE zFactorCorrectedNearSurface;
    L3DPRX_G1_ZFACTORCORRECTEDDPR zFactorCorrectedDPR;
    L3DPRX_G1_ZFACTORCORRECTEDESURFACEDPR zFactorCorrectedESurfaceDPR;
    L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACEDPR zFactorCorrectedNearSurfaceDPR;
    L3DPRX_G1_ZFACTORMEASURED zFactorMeasured;
    L3DPRX_G1_DM dm;
    L3DPRX_G1_DBNW dBnw;
    L3DPRX_G1_EPSILONDR epsilonDPR;
    L3DPRX_G1_EPSILON epsilon;
    L3DPRX_G1_ZETA zeta;
    L3DPRX_G1_PIAHB piaHB;
    L3DPRX_G1_PIAHYBRID piaHybrid;
    L3DPRX_G1_PIAHYBRIDDPR piaHybridDPR;
}

#endif

#define _L3DPRX_G1_
L3DPRX_G1_PIASRT piaSRT;
L3DPRX_G1_PIASRTDPR piaSRTdpr;
L3DPRX_G1_PIAFINAL piaFinal;
L3DPRX_G1_PIAFINALDPR piaFinalDPR;
L3DPRX_G1_PIAFINALSUBSET piaFinalSubset;
L3DPRX_G1_PIAFINALDPRSUBSET piaFinalDPRsubset;
L3DPRX_G1_HEIGHTBB heightBB;
L3DPRX_G1_HEIGHTBBNADIR heightBBnadir;
L3DPRX_G1_BBWIDTHNADIR BBwidthNadir;
L3DPRX_G1_HEIGHTSTORMTOP heightStormTop;
L3DPRX_G1_BBWIDTH BBwidth;
L3DPRX_G1_OBSERVATIONCOUNTS observationCounts;
L3DPRX_G1_PRECIPRATELOCALTIME precipRateLocalTime;
L3DPRX_G1_DFRMNEARSURFACE DFRmNearSurface;
L3DPRX_G1_DFRNEARSURFACE DFRNearSurface;
float precipRateNearSurfaceUnconditional[7][72][28];
float precipProbabilityNearSurface[7][72][28];
}

#endif

ifndef _L3DPRX_GRIDS_
define _L3DPRX_GRIDS_

typedef struct {
L3DPRX_G1 G1;
L3DPRX_G2 G2;
}
L3DPRX_GRIDS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L3DPRX_G2_DFRNEARSURFACE/
INTEGER*4 count(536,1440,3)
REAL*4 mean(536,1440,3)
REAL*4 stdev(536,1440,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_DFRMNEARSURFACE/
INTEGER*4 count(536,1440,3)
REAL*4 mean(536,1440,3)
REAL*4 stdev(536,1440,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_OBSERVATIONCOUNTS/
    INTEGER*4 total(536,1440,7)
    INTEGER*4 pia(536,1440,7,7)
    INTEGER*4 shallowRain(536,1440,7)
END STRUCTURE

STRUCTURE /L3DPRX_G2_BBWIDTH/
    INTEGER*4 count(536,1440,7,3)
    REAL*4 mean(536,1440,7,3)
    REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_HEIGHTSTORMTOP/
    INTEGER*4 count(536,1440,7,3)
    REAL*4 mean(536,1440,7,3)
    REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_HEIGHTBB/
    INTEGER*4 count(536,1440,7,3)
    REAL*4 mean(536,1440,7,3)
    REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PIAFINALDPR/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
    REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PIAFINAL/
    INTEGER*4 count(536,1440,7,7,3)
    REAL*4 mean(536,1440,7,7,3)
    REAL*4 stdev(536,1440,7,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PIASRTDPR/
    INTEGER*4 count(536,1440,4,7,3)
    REAL*4 mean(536,1440,4,7,3)
END STRUCTURE
REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2.PIASRT/
  INTEGER*4 count(536,1440,7,7,3)
  REAL*4 mean(536,1440,7,7,3)
  REAL*4 stdev(536,1440,7,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2.PIAHYBRI.DPR/
  INTEGER*4 count(536,1440,4,7,3)
  REAL*4 mean(536,1440,4,7,3)
  REAL*4 stdev(536,1440,4,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2.PIAHYBRID/
  INTEGER*4 count(536,1440,7,7,3)
  REAL*4 mean(536,1440,7,7,3)
  REAL*4 stdev(536,1440,7,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2.PIAHB/
  INTEGER*4 count(536,1440,7,7,3)
  REAL*4 mean(536,1440,7,7,3)
  REAL*4 stdev(536,1440,7,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2.ZETA/
  INTEGER*4 count(536,1440,7,7,3)
  REAL*4 mean(536,1440,7,7,3)
  REAL*4 stdev(536,1440,7,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2.EPSILON/
  INTEGER*4 count(536,1440,7,3)
  REAL*4 mean(536,1440,7,3)
  REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2.EPSILON.DPR/
  INTEGER*4 count(536,1440,4,5,3)
  REAL*4 mean(536,1440,4,5,3)
  REAL*4 stdev(536,1440,4,5,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_DBNW/
  INTEGER*4 count(536, 1440, 7, 5, 3)
  REAL*4 mean(536, 1440, 7, 5, 3)
  REAL*4 stdev(536, 1440, 7, 5, 3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_DM/
  INTEGER*4 count(536, 1440, 7, 5, 3)
  REAL*4 mean(536, 1440, 7, 5, 3)
  REAL*4 stdev(536, 1440, 7, 5, 3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_ZFACTORMEASURED/
  INTEGER*4 count(536, 1440, 7, 5, 3)
  REAL*4 mean(536, 1440, 7, 5, 3)
  REAL*4 stdev(536, 1440, 7, 5, 3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACEDPR/
  INTEGER*4 count(536, 1440, 4, 3)
  REAL*4 mean(536, 1440, 4, 3)
  REAL*4 stdev(536, 1440, 4, 3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_ZFACTORCORRECTEDDESURFACEDPR/
  INTEGER*4 count(536, 1440, 4, 3)
  REAL*4 mean(536, 1440, 4, 3)
  REAL*4 stdev(536, 1440, 4, 3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_ZFACTORCORRECTEDDPR/
  INTEGER*4 count(536, 1440, 4, 5, 3)
  REAL*4 mean(536, 1440, 4, 5, 3)
  REAL*4 stdev(536, 1440, 4, 5, 3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_ZFACTORMEASUREDNEARSURFACE/
  INTEGER*4 count(536, 1440, 7, 3)
  REAL*4 mean(536, 1440, 7, 3)
  REAL*4 stdev(536, 1440, 7, 3)
END STRUCTURE
STRUCTURE /L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACE/
    INTEGER*4 count(536,1440,7,3)
    REAL*4 mean(536,1440,7,3)
    REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_ZFACTORCORRECTEDDESURFACE/
    INTEGER*4 count(536,1440,7,3)
    REAL*4 mean(536,1440,7,3)
    REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_ZFACTORCORRECTED/
    INTEGER*4 count(536,1440,7,5,3)
    REAL*4 mean(536,1440,7,5,3)
    REAL*4 stdev(536,1440,7,5,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PRECIPRATEAVE24/
    INTEGER*4 count(536,1440,7,3)
    REAL*4 mean(536,1440,7,3)
    REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PRECIPICEINTEGRATED/
    INTEGER*4 count(536,1440,7,3)
    REAL*4 mean(536,1440,7,3)
    REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PRECIPWATERINTEGRATED/
    INTEGER*4 count(536,1440,7,3)
    REAL*4 mean(536,1440,7,3)
    REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_MIXEDPHRATENEARSURFACE/
    INTEGER*4 count(536,1440,7,3)
    REAL*4 mean(536,1440,7,3)
    REAL*4 stdev(536,1440,7,3)
END STRUCTURE
STRUCTURE /L3DPRX_G2_SNOWRATENEARSURFACE/
   INTEGER*4 count(536,1440,7,3)
   REAL*4 mean(536,1440,7,3)
   REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_RAINRATENEARSURFACE/
   INTEGER*4 count(536,1440,7,3)
   REAL*4 mean(536,1440,7,3)
   REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PRECIPRATENEARSURFACE/
   INTEGER*4 count(536,1440,7,3)
   REAL*4 mean(536,1440,7,3)
   REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PRECIPRATEESURFACE2/
   INTEGER*4 count(536,1440,7,3)
   REAL*4 mean(536,1440,7,3)
   REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PRECIPRATEESURFACE/
   INTEGER*4 count(536,1440,7,3)
   REAL*4 mean(536,1440,7,3)
   REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_MIXEDPHRATE/
   INTEGER*4 count(536,1440,7,5,3)
   REAL*4 mean(536,1440,7,5,3)
   REAL*4 stdev(536,1440,7,5,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_FLAGHEAVYICEPRECIP/
   INTEGER*4 count(536,1440,7,3)
   REAL*4 mean(536,1440,7,3)
   REAL*4 stdev(536,1440,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_SNOWRATE/
INTEGER*4 count(536,1440,7,5,3)
REAL*4 mean(536,1440,7,5,3)
REAL*4 stdev(536,1440,7,5,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_RAINRATE/
  INTEGER*4 count(536,1440,7,5,3)
  REAL*4 mean(536,1440,7,5,3)
  REAL*4 stdev(536,1440,7,5,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2_PRECIPRATE/
  INTEGER*4 count(536,1440,7,5,3)
  REAL*4 mean(536,1440,7,5,3)
  REAL*4 stdev(536,1440,7,5,3)
END STRUCTURE

STRUCTURE /L3DPRX_G2/
  RECORD /L3DPRX_G2_PRECIPRATE/ precipRate
  RECORD /L3DPRX_G2_RAINRATE/ rainRate
  RECORD /L3DPRX_G2_SNOWRATE/ snowRate
  RECORD /L3DPRX_G2_FLAGHEAVYICEPRECIP/ flagHeavyIcePrecip
  RECORD /L3DPRX_G2_MIXEDPHRATE/ mixedPhRate
  RECORD /L3DPRX_G2_PRECIPRATEESURFACE/ precipRateESurface
  RECORD /L3DPRX_G2_PRECIPRATEESURFACE2/ precipRateESurface2
  RECORD /L3DPRX_G2_PRECIPRATENEARSURFACE/ precipRateNearSurface
  RECORD /L3DPRX_G2_RAINRATENEARSURFACE/ rainRateNearSurface
  RECORD /L3DPRX_G2_SNOWRATENEARSURFACE/ snowRateNearSurface
  RECORD /L3DPRX_G2_MIXEDPHRATENEARSURFACE/ mixedPhRateNearSurface
  RECORD /L3DPRX_G2_PRECIPWATERINTEGRATED/ precipWaterIntegrated
  RECORD /L3DPRX_G2_PRECIPICEINTEGRATED/ precipIceIntegrated
  RECORD /L3DPRX_G2_PRECIPRATEAVE24/ precipRateAve24
  RECORD /L3DPRX_G2_ZFACTORCORRECTED/ zFactorCorrected
  RECORD /L3DPRX_G2_ZFACTORCORRECTEDESURFACE/ zFactorCorrectedESurface
  RECORD /L3DPRX_G2_ZFACTORCORRECTEDNEARSURFACE/ zFactorCorrectedNearSurface
  RECORD /L3DPRX_G2_ZFACTORMEASUREDNEARSURFACE/ zFactorMeasuredNearSurface
  RECORD /L3DPRX_G2_ZFACTORCORRECTEDDPR/ zFactorCorrectedDPR
  RECORD /L3DPRX_G2_ZFACTORCORRECTEDESURFACEDPR/ zFactorCorrectedESurfaceDPR
  RECORD /L3DPRX_G2_ZFACTORMEASURED/ zFactorMeasured
  RECORD /L3DPRX_G2_DM/ dm
  RECORD /L3DPRX_G2_DBNW/ dBNw
  RECORD /L3DPRX_G2_EPSILONDPR/ epsilonDPR
RECORD /L3DPRX_G2_EPSILON/ epsilon
RECORD /L3DPRX_G2_ZETA/ zeta
RECORD /L3DPRX_G2_PIAHB/ piaHB
RECORD /L3DPRX_G2_PIAHYBRID/ piaHybrid
RECORD /L3DPRX_G2_PIAHYBRIDDPR/ piaHybridDPR
RECORD /L3DPRX_G2_PIAASRT/ piaSRT
RECORD /L3DPRX_G2_PIASRTDPR/ piaSRTdpr
RECORD /L3DPRX_G2_PIAFINAL/ piaFinal
RECORD /L3DPRX_G2_PIAFINALDPR/ piaFinalDPR
RECORD /L3DPRX_G2_HEIGHTBB/ heightBB
RECORD /L3DPRX_G2_HEIGHTSTORMTOP/ heightStormTop
RECORD /L3DPRX_G2_BBWIDTH/ BBwidth
RECORD /L3DPRX_G2_OBSERVATIONCOUNTS/ observationCounts
RECORD /L3DPRX_G2_DFRMNEARSURFACE/ DFRmNearSurface
RECORD /L3DPRX_G2_DFRNEARSURFACE/ DFRNearSurface
REAL*4 precipRateNearSurfaceUnconditional(536,1440,7)
REAL*4 precipProbabilityNearSurface(536,1440,7)
END STRUCTURE

STRUCTURE /L3DPRX_G1_DFRNEARSURFACE/
  INTEGER*4 count(28,72,3,3)
  REAL*4 mean(28,72,3,3)
  REAL*4 stdev(28,72,3,3)
  INTEGER*4 hist(28,72,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_DFRMNEARSURFACE/
  INTEGER*4 count(28,72,3,3)
  REAL*4 mean(28,72,3,3)
  REAL*4 stdev(28,72,3,3)
  INTEGER*4 hist(28,72,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PRECIPRATELOCALTIME/
  INTEGER*4 count(28,72,7,24,3)
  REAL*4 mean(28,72,7,24,3)
  REAL*4 stdev(28,72,7,24,3)
END STRUCTURE

STRUCTURE /L3DPRX_G1_OBSERVATIONCOUNTS/
  INTEGER*4 total(28,72,7,3)
  INTEGER*4 localTime(28,72,7,24,3)
  INTEGER*4 pia(28,72,7,7,3)
INTEGER*4 shallowRain(28,72,7,3)
END STRUCTURE

STRUCTURE /L3DPRX_G1_BBWIDTH/
    INTEGER*4 count(28,72,7,3,3)
    REAL*4 mean(28,72,7,3,3)
    REAL*4 stdev(28,72,7,3,3)
    INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_HEIGHTSTORMTOP/
    INTEGER*4 count(28,72,7,3,3)
    REAL*4 mean(28,72,7,3,3)
    REAL*4 stdev(28,72,7,3,3)
    INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_BBWIDTHNADIR/
    INTEGER*4 count(28,72,7,3,3)
    REAL*4 mean(28,72,7,3,3)
    REAL*4 stdev(28,72,7,3,3)
    INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_HEIGHTBBNADIR/
    INTEGER*4 count(28,72,7,3,3)
    REAL*4 mean(28,72,7,3,3)
    REAL*4 stdev(28,72,7,3,3)
    INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_HEIGHTBB/
    INTEGER*4 count(28,72,7,3,3)
    REAL*4 mean(28,72,7,3,3)
    REAL*4 stdev(28,72,7,3,3)
    INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PIAFINALDPRSUBSET/
    INTEGER*4 count(28,72,4,7,3,3)
    REAL*4 mean(28,72,4,7,3,3)
    REAL*4 stdev(28,72,4,7,3,3)
    INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PIAFINALSUBSET/
   INTEGER*4 count(28,72,7,7,3,3)
   REAL*4 mean(28,72,7,7,3,3)
   REAL*4 stdev(28,72,7,7,3,3)
   INTEGER*4 hist(28,72,7,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PIAFINALDPR/
   INTEGER*4 count(28,72,4,7,3,3)
   REAL*4 mean(28,72,4,7,3,3)
   REAL*4 stdev(28,72,4,7,3,3)
   INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PIAFINAL/
   INTEGER*4 count(28,72,7,7,3,3)
   REAL*4 mean(28,72,7,7,3,3)
   REAL*4 stdev(28,72,7,7,3,3)
   INTEGER*4 hist(28,72,7,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PIASRTDPR/
   INTEGER*4 count(28,72,4,7,3,3)
   REAL*4 mean(28,72,4,7,3,3)
   REAL*4 stdev(28,72,4,7,3,3)
   INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PIASRT/
   INTEGER*4 count(28,72,7,7,3,3)
   REAL*4 mean(28,72,7,7,3,3)
   REAL*4 stdev(28,72,7,7,3,3)
   INTEGER*4 hist(28,72,7,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PIAHYBRIDDPR/
   INTEGER*4 count(28,72,4,7,3,3)
   REAL*4 mean(28,72,4,7,3,3)
   REAL*4 stdev(28,72,4,7,3,3)
   INTEGER*4 hist(28,72,4,7,3,3,30)
END STRUCTURE
STRUCTURE /L3DPRX_G1_PIAHYBRID/
  INTEGER*4 count(28,72,7,7,3,3)
  REAL*4 mean(28,72,7,7,3,3)
  REAL*4 stdev(28,72,7,7,3,3)
  INTEGER*4 hist(28,72,7,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PIAHB/
  INTEGER*4 count(28,72,7,7,3,3)
  REAL*4 mean(28,72,7,7,3,3)
  REAL*4 stdev(28,72,7,7,3,3)
  INTEGER*4 hist(28,72,7,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_ZETA/
  INTEGER*4 count(28,72,7,7,3,3)
  REAL*4 mean(28,72,7,7,3,3)
  REAL*4 stdev(28,72,7,7,3,3)
  INTEGER*4 hist(28,72,7,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_EPSILON/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_EPSILONDPR/
  INTEGER*4 count(28,72,4,5,3,3)
  REAL*4 mean(28,72,4,5,3,3)
  REAL*4 stdev(28,72,4,5,3,3)
  INTEGER*4 hist(28,72,4,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_DBNW/
  INTEGER*4 count(28,72,7,5,3,3)
  REAL*4 mean(28,72,7,5,3,3)
  REAL*4 stdev(28,72,7,5,3,3)
  INTEGER*4 hist(28,72,7,5,3,3,30)
END STRUCTURE
STRUCTURE /L3DPRX_G1_DM/
    INTEGER*4 count(28,72,7,5,3,3)
    REAL*4 mean(28,72,7,5,3,3)
    REAL*4 stdev(28,72,7,5,3,3)
    INTEGER*4 hist(28,72,7,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_ZFACTORMEASURED/
    INTEGER*4 count(28,72,7,5,3,3)
    REAL*4 mean(28,72,7,5,3,3)
    REAL*4 stdev(28,72,7,5,3,3)
    INTEGER*4 hist(28,72,7,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACEDPR/
    INTEGER*4 count(28,72,4,3,3,3)
    REAL*4 mean(28,72,4,3,3,3)
    REAL*4 stdev(28,72,4,3,3,3)
    INTEGER*4 hist(28,72,4,3,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACE/
    INTEGER*4 count(28,72,7,3,3,3)
    REAL*4 mean(28,72,7,3,3,3)
    REAL*4 stdev(28,72,7,3,3,3)
    INTEGER*4 hist(28,72,7,3,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_ZFACTORCORRECTEDDPR/
    INTEGER*4 count(28,72,4,5,3,3,3)
    REAL*4 mean(28,72,4,5,3,3,3)
    REAL*4 stdev(28,72,4,5,3,3,3)
    INTEGER*4 hist(28,72,4,5,3,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_ZFACTORMEASUREDNEARSURFACE/
    INTEGER*4 count(28,72,7,3,3,3)
    REAL*4 mean(28,72,7,3,3,3)
    REAL*4 stdev(28,72,7,3,3,3)
    INTEGER*4 hist(28,72,7,3,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACE/
INTEGER*4 count(28,72,7,3,3)
REAL*4 mean(28,72,7,3,3)
REAL*4 stdev(28,72,7,3,3)
INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_ZFACTORCORRECTEDESURFACE/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_ZFACTORCORRECTED/
  INTEGER*4 count(28,72,7,5,3,3)
  REAL*4 mean(28,72,7,5,3,3)
  REAL*4 stdev(28,72,7,5,3,3)
  INTEGER*4 hist(28,72,7,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PRECIPRATEAVE24/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PRECIPICEINTEGRATED/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PRECIPWATERINTEGRATED/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_MIXEDPHRATENEARSURFACE/
  INTEGER*4 count(28,72,7,3,3)
REAL*4 mean(28,72,7,3,3)
REAL*4 stdev(28,72,7,3,3)
INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_SNOWRATENEARSURFACE/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_RAINRATENEARSURFACE/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PRECIPRATENEARSURFACE/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PRECIPRATEESURFACE2/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PRECIPRATEESURFACE/
  INTEGER*4 count(28,72,7,3,3)
  REAL*4 mean(28,72,7,3,3)
  REAL*4 stdev(28,72,7,3,3)
  INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_MIXEDPHRATE/
  INTEGER*4 count(28,72,7,5,3,3)
  REAL*4 mean(28,72,7,5,3,3)
REAL*4 stdev(28,72,7,5,3,3)
INTEGER*4 hist(28,72,7,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_FLAGHEAVYICEPRECIP/
   INTEGER*4 count(28,72,7,3,3)
   REAL*4 mean(28,72,7,3,3)
   REAL*4 stdev(28,72,7,3,3)
   INTEGER*4 hist(28,72,7,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_SNOWRATE/
   INTEGER*4 count(28,72,7,5,3,3)
   REAL*4 mean(28,72,7,5,3,3)
   REAL*4 stdev(28,72,7,5,3,3)
   INTEGER*4 hist(28,72,7,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_RAINRATE/
   INTEGER*4 count(28,72,7,5,3,3)
   REAL*4 mean(28,72,7,5,3,3)
   REAL*4 stdev(28,72,7,5,3,3)
   INTEGER*4 hist(28,72,7,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1_PRECIPRATE/
   INTEGER*4 count(28,72,7,5,3,3)
   REAL*4 mean(28,72,7,5,3,3)
   REAL*4 stdev(28,72,7,5,3,3)
   INTEGER*4 hist(28,72,7,5,3,3,30)
END STRUCTURE

STRUCTURE /L3DPRX_G1/
   RECORD /L3DPRX_G1_PRECIPRATE/ precipRate
   RECORD /L3DPRX_G1_RAINRATE/ rainRate
   RECORD /L3DPRX_G1_SNOWRATE/ snowRate
   RECORD /L3DPRX_G1_FLAGHEAVYICEPRECIP/ flagHeavyIcePrecip
   RECORD /L3DPRX_G1_MIXEDPHRATE/ mixedPhRate
   RECORD /L3DPRX_G1_PRECIPRATEESURFACE/ precipRateESurface
   RECORD /L3DPRX_G1_PRECIPRATEESURFACE2/ precipRateESurface2
   RECORD /L3DPRX_G1_PRECIPRATENEARSURFACE/ precipRateNearSurface
   RECORD /L3DPRX_G1_RAINRATENEARSURFACE/ rainRateNearSurface
   RECORD /L3DPRX_G1_SNOWRATENEARSURFACE/ snowRateNearSurface
RECORD /L3DPRX_G1_MIXEDPHRATENEARSURFACE/ mixedPhRateNearSurface
RECORD /L3DPRX_G1_PRECIPWATERINTEGRATED/ precipWaterIntegrated
RECORD /L3DPRX_G1_PRECIPICEINTEGRATED/ precipIceIntegrated
RECORD /L3DPRX_G1_PRECIPRATEAVE24/ precipRateAve24
RECORD /L3DPRX_G1_ZFACTORCORRECTED/ zFactorCorrected
RECORD /L3DPRX_G1_ZFACTORCORRECTEDESURFACE/ zFactorCorrectedESurface
RECORD /L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACE/ zFactorCorrectedNearSurface
RECORD /L3DPRX_G1_ZFACTORMEASUREDNearsurface/ zFactorMeasuredNearSurface
RECORD /L3DPRX_G1_ZFACTORCORRECTEDDPR/ zFactorCorrectedDPR
RECORD /L3DPRX_G1_ZFACTORCORRECTEDESURFACE_DPR/ zFactorCorrectedESurfaceDPR
RECORD /L3DPRX_G1_ZFACTORCORRECTEDNEARSURFACE_DPR/ zFactorCorrectedNearSurfaceDPR
RECORD /L3DPRX_G1_ZFACTORMEASURED/ zFactorMeasured
RECORD /L3DPRX_G1_DM/ dm
RECORD /L3DPRX_G1_DBNW/ dBNw
RECORD /L3DPRX_G1_EPSILON_DPR/ epsilonDPR
RECORD /L3DPRX_G1_EPSILON/ epsilon
RECORD /L3DPRX_G1_ZETA/ zeta
RECORD /L3DPRX_G1_PIAHB/ piaHB
RECORD /L3DPRX_G1_PIAHYBRID/ piaHybrid
RECORD /L3DPRX_G1_PIAHYBRID_DPR/ piaHybridDPR
RECORD /L3DPRX_G1_PIASRT/ piaSRT
RECORD /L3DPRX_G1_PIASRT_DPR/ piaSRTDPR
RECORD /L3DPRX_G1_PIAFINAL/ piaFinal
RECORD /L3DPRX_G1_PIAFINAL_DPR/ piaFinalDPR
RECORD /L3DPRX_G1_PIAFINAL_SUBSET/ piaFinalSubset
RECORD /L3DPRX_G1_PIAFINAL_DPR_SUBSET/ piaFinalDPRsubset
RECORD /L3DPRX_G1_HEIGHTBB/ heightBB
RECORD /L3DPRX_G1_HEIGHTBB_NADIR/ heightBBnadir
RECORD /L3DPRX_G1_BBWIDTH_NADIR/ BBwidthNadir
RECORD /L3DPRX_G1_HEIGHTSTORMTOP/ heightStormTop
RECORD /L3DPRX_G1_BBWIDTH/ BBwidth
RECORD /L3DPRX_G1_OBSERVATION_COUNTS/ observationCounts
RECORD /L3DPRX_G1_PRECIP_RATE_LOCAL_TIME/ precipRateLocalTime
RECORD /L3DPRX_G1_DRMNEARSURFACE/ DFRmNearSurface
RECORD /L3DPRX_G1_DFRNEARSURFACE/ DFRNearSurface
REAL*4 precipRateNearSurfaceUnconditional(28,72,7)
REAL*4 precipProbabilityNearSurface(28,72,7)

END STRUCTURE

STRUCTURE /L3DPRX_GRIDS/
  RECORD /L3DPRX_G1/ G1
  RECORD /L3DPRX_G2/ G2
END STRUCTURE
5.70 2BCMBX - Level-2 DPR and GMI Combined

The Combined Level-2 product, 2BCMBX, "Level-2 DPR and GMI Combined," is written as a two-swath structure. The first swath, NS, contains 49 rays that match Ku DPR. The second swath, FS, contains 49 rays that match Ka Matched DPR. Surface variables refer to the level of the 2ADPR "near surface", not the "estimated surface". The following sections describe the structure and contents of the format.

Dimension definitions:

- nsrn var Number of scans in the granule.
- nrayNS 49 Number of rays (angle bins) in each NS scan.
- nrayFS 49 Number of rays (angle bins) in each FS scan.
- nBnEnv 10 Number of environmental bins.
- nBnPSDlo 9 Number of low resolution vertical range bins. The bin indices of the low resolution PSD profile parameters are found in PSDparamLowNode.
- nBnPSDhi 88 Number of high resolution vertical range bins at 250m interval.
- nPSDlo 2 Number of low resolution precipitation drop-size distribution parameters. Parameters are log10(Nw), mu.
- nPSDhi 1 Number of high resolution precipitation drop-size distribution parameters.
- nBnTrBnd 2 Number of bins in phase transition boundary.
- nBnTr 10 Number of bins in phase transition.
- nPhsBnN 5 Number of phase bin nodes.
- nAB 2 Number of power law parameters. These parameters describe particle density. The parameters are alpha and beta.
- nemiss 13 Number of microwave surface emissivities for GMI channels, including separate emissivities for the double side-band channels.
- nKuKa 2 Number of Ku and Ka
- ncomp 5 Maximum number principal components (prinComp) stored for a given observed reflectivity profile.

Figure 1073 through Figure 1089 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 1073: Data Format Structure for 2BCMBX, Level-2 DPR and GMI Combined
<table>
<thead>
<tr>
<th>Field</th>
<th>Size</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>NS_SwathHeader</td>
<td>19 bytes</td>
<td>Metadata</td>
</tr>
<tr>
<td>ScanTime</td>
<td>19 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>Latitude</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>Longitude</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>scanStatus</td>
<td>24 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>navigation</td>
<td>84 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>Input</td>
<td>3038 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>aPriori</td>
<td>3136 bytes</td>
<td>Group: nscan</td>
</tr>
<tr>
<td>surfaceAirPressure</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>surfaceAirTemperature</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>surfaceVaporDensity</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>skinTemperature</td>
<td>4 bytes</td>
<td>Array: nrayNS x nscan</td>
</tr>
<tr>
<td>envParamNode</td>
<td>2 bytes</td>
<td>Array: nBnEnv x nrayNS x nscan</td>
</tr>
<tr>
<td>airPressure</td>
<td>4 bytes</td>
<td>Array: nBnEnv x nrayNS x nscan</td>
</tr>
<tr>
<td>airTemperature</td>
<td>4 bytes</td>
<td>Array: nBnEnv x nrayNS x nscan</td>
</tr>
<tr>
<td>vaporDensity</td>
<td>4 bytes</td>
<td>Array: nBnEnv x nrayNS x nscan</td>
</tr>
<tr>
<td>cloudLiqWaterCont</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>cloudIceWaterCont</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>phaseBinNodes</td>
<td>2 bytes</td>
<td>Array: nPhsBnN x nrayNS x nscan</td>
</tr>
<tr>
<td>PSDparamLowNode</td>
<td>2 bytes</td>
<td>Array: nBnPSDio x nrayNS x nscan</td>
</tr>
<tr>
<td>precipTotPSDparamLow</td>
<td>4 bytes</td>
<td>Array: nPSDlo x nBnPSDlo x nrayNS x nscan</td>
</tr>
<tr>
<td>precipTotPSDparamHigh</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>precipTotWaterCont</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
<tr>
<td>precipTotWaterContSigma</td>
<td>4 bytes</td>
<td>Array: nBnPSDhi x nrayNS x nscan</td>
</tr>
</tbody>
</table>

continued on next figure

Figure 1074: Data Format Structure for 2BCMBX, NS, NS
continued from last figure

- precipTotRate 4 bytes Array: nBnPSDhi x nrayNS x nscan
- precipTotRateSigma 4 bytes Array: nBnPSDhi x nrayNS x nscan
- liqMassFracTrans 4 bytes Array: nBnTr x nrayNS x nscan
- liqRateFracTrans 4 bytes Array: nBnTr x nrayNS x nscan
- surfPrecipTotRate 4 bytes Array: nrayNS x nscan
- surfPrecipTotRateSigma 4 bytes Array: nrayNS x nscan
- surfLiqRateFrac 4 bytes Array: nrayNS x nscan
- tenMeterWindSpeed 4 bytes Array: nrayNS x nscan
- surfEmissivity 4 bytes Array: nemiss x nrayNS x nscan
- simulatedBrightTemp 4 bytes Array: nemiss x nrayNS x nscan
- nubfPIAfactor 4 bytes Array: nrayNS x nscan
- multiScatMaxContrib 4 bytes Array: nrayNS x nscan
- surfEmissSigma 4 bytes Array: nemiss x nrayNS x nscan
- tenMeterWindSigma 4 bytes Array: nrayNS x nscan
- skinTempSigma 4 bytes Array: nrayNS x nscan
- columnVaporSigma 4 bytes Array: nrayNS x nscan
- columnCloudLiqSigma 4 bytes Array: nrayNS x nscan
- errorOfDataFit 4 bytes Array: nrayNS x nscan
- pia 4 bytes Array: nrayNS x nscan
- CorrectedReflectFactor 4 bytes Array: nBnPSDhi x nrayNS x nscan
- FLG 588 bytes Group: nscan

Figure 1075: Data Format Structure for 2BCMBX, NS
Figure 1076: Data Format Structure for 2BCMBX, FS, FS
continued from last figure

- precipTotRate 4 bytes, Array: nBuPSDhi x nrayFS x nscan
- precipTotRateSigma 4 bytes, Array: nBuPSDhi x nrayFS x nscan
- liqMassFracTrans 4 bytes, Array: nBuTr x nrayFS x nscan
- liqRateFracTrans 4 bytes, Array: nBuTr x nrayFS x nscan
- surfPrecipTotRate 4 bytes, Array: nrayFS x nscan
- surfPrecipTotRateSigma 4 bytes, Array: nrayFS x nscan
- surfLiqRateFrac 4 bytes, Array: nrayFS x nscan
- tenMeterWindSpeed 4 bytes, Array: nrayFS x nscan
- surfEmissivity 4 bytes, Array: nemiss x nrayFS x nscan
- simulatedBrightTemp 4 bytes, Array: nemiss x nrayFS x nscan
- nubPIAfactor 4 bytes, Array: nrayFS x nscan
- multiScatMaxContrib 4 bytes, Array: nrayFS x nscan
- surfEmissSigma 4 bytes, Array: nemiss x nrayFS x nscan
- tenMeterWindSigma 4 bytes, Array: nrayFS x nscan
- skinTempSigma 4 bytes, Array: nrayFS x nscan
- columnVaporSigma 4 bytes, Array: nrayFS x nscan
- columnCloudLiqSigma 4 bytes, Array: nrayFS x nscan
- errorOfDataFit 4 bytes, Array: nrayFS x nscan
- pia 4 bytes, Array: nKuKa x nrayFS x nscan
- correctedReflectFactor 4 bytes, Array: nKuKa x nBuPSDhi x nrayFS x nscan
- FLG 588 bytes, Group: nscan

Figure 1077: Data Format Structure for 2BCMBX, FS
Figure 1078: Data Format Structure for 2BCMBX, NS, ScanTime

Figure 1079: Data Format Structure for 2BCMBX, NS, scanStatus
Figure 1080: Data Format Structure for 2BCMBX, NS, navigation
5.70 2BCMBX - Level-2 DPR and GMI Combined

Figure 1081: Data Format Structure for 2BCMBX, NS, Input

- `surfaceElevation` 4 bytes Array: nrayNS x nscan
- `surfaceType` 4 bytes Array: nrayNS x nscan
- `localZenithAngle` 4 bytes Array: nrayNS x nscan
- `precipitationFlag` 4 bytes Array: nrayNS x nscan
- `surfaceRangeBin` 2 bytes Array: nrayNS x nscan
- `lowestClutterFreeBin` 2 bytes Array: nrayNS x nscan
- `ellipsoidBinOffset` 4 bytes Array: nrayNS x nscan
- `stormTopBin` 2 bytes Array: nrayNS x nscan
- `stormTopAltitude` 4 bytes Array: nrayNS x nscan
- `zeroDegBin` 2 bytes Array: nrayNS x nscan
- `zeroDegAltitude` 4 bytes Array: nrayNS x nscan
- `precipitationType` 4 bytes Array: nrayNS x nscan
- `precipTypeQualityFlag` 4 bytes Array: nrayNS x nscan
- `piaEffective` 4 bytes Array: nrayNS x nscan
- `piaEffectiveSigma` 4 bytes Array: nrayNS x nscan
- `piaEffectiveReliabFlag` 2 bytes Array: nrayNS x nscan
- `sigmaZeroMeasured` 4 bytes Array: nrayNS x nscan
- `snowIceCover` 4 bytes Array: nrayNS x nscan

Figure 1082: Data Format Structure for 2BCMBX, NS, aPriori

- `profClass` 4 bytes Array: nrayNS x nscan
- `prinComp` 4 bytes Array: ncomp x nrayNS x nscan
- `surfPrecipBiasRatio` 4 bytes Array: nrayNS x nscan
- `initNw` 4 bytes Array: nBnPSDlo x nrayNS x nscan

Figure 1083: Data Format Structure for 2BCMBX, NS, FLG

- `ioQuality` 4 bytes Array: nrayNS x nscan
- `multiScatCalc` 4 bytes Array: nrayNS x nscan
- `algoType` 4 bytes Array: nrayNS x nscan
Figure 1084: Data Format Structure for 2BCMBX, FS, ScanTime

Figure 1085: Data Format Structure for 2BCMBX, FS, scanStatus
Figure 1086: Data Format Structure for 2BCMBX, FS, navigation
Figure 1087: Data Format Structure for 2BCMBX, FS, Input

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.
5.70 2BCMBX - Level-2 DPR and GMI Combined

Figure 1088: Data Format Structure for 2BCMBX, FS, aPriori

Figure 1089: Data Format Structure for 2BCMBX, FS, FLG

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**NS** (Swath)

**NS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in NS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value

Latitude (4-byte float, array size: nrayNS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nrayNS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scanStatus (Group in NS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>
**dataWarning** (1-byte integer, array size: nscan):
Flag of data warning for each scan.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Beam matching is abnormal</td>
</tr>
<tr>
<td>1</td>
<td>VPRF table is abnormal</td>
</tr>
<tr>
<td>2</td>
<td>Surface table is abnormal</td>
</tr>
<tr>
<td>3</td>
<td>geoWarning is not zero</td>
</tr>
<tr>
<td>4</td>
<td>Operational mode is not observation mode</td>
</tr>
<tr>
<td>5</td>
<td>GPS status is abnormal</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Check sum of L1A is abnormal</td>
</tr>
</tbody>
</table>

**missing** (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Scan is missing</td>
</tr>
<tr>
<td>1</td>
<td>Science telemetry packet missing</td>
</tr>
<tr>
<td>2</td>
<td>Science telemetry segment within packet missing</td>
</tr>
<tr>
<td>3</td>
<td>Science telemetry other missing</td>
</tr>
<tr>
<td>4</td>
<td>Housekeeping (HK) telemetry packet missing</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

**modeStatus** (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>1</td>
<td>SCorientation not 0 or 180</td>
</tr>
<tr>
<td>2</td>
<td>pointingStatus not 0</td>
</tr>
<tr>
<td>3</td>
<td>Non-routine limitErrorFlag</td>
</tr>
<tr>
<td>4</td>
<td>Non-routine operationalMode (not 1 or 11)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality.
A zero integer value of geoError indicates 'good' geolocation. A non-zero value broken
down into the bit flags below indicates the specified reason, where bit 0 is the least
significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).
Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the
reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of
these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero,
so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to
the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

geoWarning (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in
dataQuality. Warnings indicate unusual conditions. These conditions do not indicate
bad geolocation but are flagged as a warning that further review of the data may be
useful. A zero integer value indicates usual geolocation. A non-zero value broken down
into the following bit flags indicates the following, where bit 0 is the least significant bit
(i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
</tbody>
</table>
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

SCrientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SConcernation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
</tbody>
</table>
2562 5 STANDARD GPM PRODUCTS

2 SUNPOINT
3 GSPM (Gyro-less Sun Point)
4 MSM (Mission Science Mode)
5 SLEW
6 DELTAH
7 DELTAV
-99 UNKNOWN -- ACS mode unavailable

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
</tbody>
</table>
limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks.
limitErrorFlag may be used in modeStatus.
Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the spacecraft is halfway through granule 10 and starting the descending half of the granule. Values range from 0 to 100000. Special values are defined as:
-9999.9 Missing value

navigation (Group in NS)

scPos (4-byte float, array size: XYZ x nscan):
The position vector (m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as:
-9999.9 Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector (m/s⁻¹) of the spacecraft in ECEF Coordinates at the Scan mid-Time. Values range from -10000000 to 10000000 m/s. Special values are defined as:
-9999.9 Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -70 to 70 degrees. Special values are defined as:
-9999.9 Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value
scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeo (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeo (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeo (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

**Input** (Group in NS)

**surfaceElevation** (4-byte float, array size: nrayNS x nscan):
Altitudes above the earth ellipsoid of the surface gates from 2AKu. Values are in m. Special values are defined as:
-9999.9 Missing value

**surfaceType** (4-byte integer, array size: nrayNS x nscan):
Surface type from 2AKu. Special values are defined as:
-9999 Missing value

**localZenithAngle** (4-byte float, array size: nrayNS x nscan):
Zenith angle of the ray at the earth’s surface from 2AKu. Values are in degree. Special values are defined as:
-9999.9 Missing value
**precipitationFlag** (4-byte integer, array size: nrayNS x nscan):
Precipitation flag from 2AKu. Special values are defined as:
-9999 Missing value

**surfaceRangeBin** (2-byte integer, array size: nrayNS x nscan):
Index of the surface range bin from 2AKu. Special values are defined as:
-9999 Missing value

**lowestClutterFreeBin** (2-byte integer, array size: nrayNS x nscan):
Index of lowest clutter-free bin from 2AKu. Special values are defined as:
-9999 Missing value

**ellipsoidBinOffset** (4-byte float, array size: nrayNS x nscan):
Offset of surface bin from the earth ellipsoid from 2AKu. Values are in m. Special values are defined as:
-9999.9 Missing value

**stormTopBin** (2-byte integer, array size: nrayNS x nscan):
Index of storm top bin from 2AKu. Special values are defined as:
-9999 Missing value

**stormTopAltitude** (4-byte float, array size: nrayNS x nscan):
Altitude of storm top bin from 2AKu. Values are in m. Special values are defined as:
-9999.9 Missing value

**zeroDegBin** (2-byte integer, array size: nrayNS x nscan):
Range bin of the freezing level. Special values are defined as:
-9999 Missing value

**zeroDegAltitude** (4-byte float, array size: nrayNS x nscan):
Altitude of the freezing level. Values are in m. Special values are defined as:
-9999.9 Missing value

**precipitationType** (4-byte integer, array size: nrayNS x nscan):
Precipitation type classification from 2AKu. Special values are defined as:
-9999 Missing value

**precipTypeQualityFlag** (4-byte integer, array size: nrayNS x nscan):
Quality flag of precipitation type from 2AKu. Special values are defined as:
-9999 Missing value

**piaEffective** (4-byte float, array size: nrayNS x nscan):
Effective 2-way PIA from 2AKu. Values are in dB. Special values are defined as:
-9999.9 Missing value

**piaEffectiveSigma** (4-byte float, array size: nrayNS x nscan):
Effective PIA uncertainty from 2AKu. Values are in dB. Special values are defined as:
-9999.9 Missing value

**piaEffectiveReliabFlag** (2-byte integer, array size: nrayNS x nscan):
Reliability flag of effective PIA from 2AKu. Special values are defined as:
-9999 Missing value
**sigmaZeroMeasured** (4-byte float, array size: nrayNS x nscan):
The surface normalized radar cross section. Values range from -40 to 42 dB. Special values are defined as:
-9999.9 Missing value

**snowIceCover** (4-byte integer, array size: nrayNS x nscan):
Snow and ice cover. Values are defined as: 0 = ice-free ocean 1 = snow-free land 2 = snow-covered land 3 = sea ice. Special values are defined as:
-9999 Missing value

**aPriori** (Group in NS)

**profClass** (4-byte integer, array size: nrayNS x nscan):
The class number of the observed reflectivity profile using a classification based upon measured reflectivity structure features. Unclassified profiles are assigned a value of -9999.

**prinComp** (4-byte float, array size: ncomp x nrayNS x nscan):
Principal components of the observed reflectivity profile, up to ncomp in number, that describe the primary modes of reflectivity structural variability. Unused principal components are assigned a value of -9999.9.

**surfPrecipBiasRatio** (4-byte float, array size: nrayNS x nscan):
The a priori ratio of mean MS-mode to NS-mode surface rain rates for the given observed reflectivity profile. Special values are defined as:
-9999.9 Missing value

**initNw** (4-byte float, array size: nBnPSDlo x nrayNS x nscan):
The initial values of the ensemble-mean, low-resolution (nBnPSDlo bins) profile of Nw associated with a given observed reflectivity profile. Nw is the intercept of the normalized gamma distribution used to describe the precipitation particle size distribution. Special values are defined as:
-9999.9 Missing value

**surfaceAirPressure** (4-byte float, array size: nrayNS x nscan):
Surface air pressure. Values range from 300 to 1100 hPa. Special values are defined as:
-9999.9 Missing value

**surfaceAirTemperature** (4-byte float, array size: nrayNS x nscan):
Surface air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

**surfaceVaporDensity** (4-byte float, array size: nrayNS x nscan):
Surface vapor density. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value
skinTemperature (4-byte float, array size: nrayNS x nscan):
Surface skin temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

envParamNode (2-byte integer, array size: nBnEnv x nrayNS x nscan):
Bin indices for environmental parameters. Special values are defined as:
-9999 Missing value

airPressure (4-byte float, array size: nBnEnv x nrayNS x nscan):
Air pressure. Values range from 50 to 1100 hPa. Special values are defined as:
-9999.9 Missing value

airTemperature (4-byte float, array size: nBnEnv x nrayNS x nscan):
Air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

vaporDensity (4-byte float, array size: nBnEnv x nrayNS x nscan):
Vapor density. Values range from 0 to 60 g/m$^3$. Special values are defined as:
-9999.9 Missing value

cloudLiqWaterCont (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Cloud liquid water content. Values range from 0 to 60 g/m$^3$. Special values are defined as:
-9999.9 Missing value

cloudIceWaterCont (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Cloud ice water content. Values range from 0 to 18 g/m$^3$. Special values are defined as:
-9999.9 Missing value

phaseBinNodes (2-byte integer, array size: nPhsBnN x nrayNS x nscan):
Bin numbers indicating (0) storm top, (1) top of mixed-phase layer, (2) maximum reflectivity in mixed-phase layer if bright band detected; otherwise, the freezing level from analysis, (3) bottom of mixed-phase layer, and (4) bottom of rain layer. Special values are defined as:
-9999 Missing value

PSDparamLowNode (2-byte integer, array size: nBnPSDlo x nrayNS x nscan):
Bin indices for low-resolution PSD parameters. Special values are defined as:
-9999 Missing value

precipTotPSDparamLow (4-byte float, array size: nPSDlo x nBnPSDlo x nrayNS x nscan):
Total precipitation low-resolution PSD parameters. Parameters are log10(Nw) with units log10(1/m$^4$) for first value of nPSDlo, mu with no units for second value.

precipTotPSDparamHigh (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation high-resolution PSD parameters. Values range from 0 to 20 mm Dm. Special values are defined as:
-9999.9 Missing value
**precipTotWaterCont** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation liquid water content. Values range from 0 to 18 g/m$^3$. Special values are defined as:
-9999.9 Missing value

**precipTotWaterContSigma** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation liquid water content uncertainty. Values range from 0 to 18 g/m$^3$. Special values are defined as:
-9999.9 Missing value

**precipTotRate** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation rate. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

**precipTotRateSigma** (4-byte float, array size: nBnPSDhi x nrayNS x nscan):
Total precipitation rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

**liqMassFracTrans** (4-byte float, array size: nBnTr x nrayNS x nscan):
Fraction of the precipitation mass that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

**liqRateFracTrans** (4-byte float, array size: nBnTr x nrayNS x nscan):
Fraction of the precipitation rate that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

**surfPrecipTotRate** (4-byte float, array size: nrayNS x nscan):
Surface rain rate. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

**surfPrecipTotRateSigma** (4-byte float, array size: nrayNS x nscan):
Surface rain rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
-99 No precipitation detected
-9999.9 Missing value

**surfLiqRateFrac** (4-byte float, array size: nrayNS x nscan):
Surface liquid precipitation rate fraction. Values range from 0 to 1. Special values are
defined as:
-9999.9  Missing value

tenMeterWindSpeed (4-byte float, array size: nrayNS x nscan):
Ten meter altitude wind speed magnitude. Values range from 0 to 100 m/s. Special values are defined as:
-9999.9  Missing value

surfEmissivity (4-byte float, array size: nemiss x nrayNS x nscan):
GMI emissivities. Values range from 0 to 1. Special values are defined as:
-9999.9  Missing value

simulatedBrightTemp (4-byte float, array size: nemiss x nrayNS x nscan):
GMI simulated brightness temperatures. Values range from 20 to 350 K. Special values are defined as:
-9999.9  Missing value

nubfPIAfactor (4-byte float, array size: nrayNS x nscan):
nubfPIAfactor is the factor applied to the Hitschfeld-Bordan path integrated attenuation to obtain the simulated path integrated attenuation, accounting for the nonuniform beamfilling by precipitation which is estimated from a 3x3 neighborhood of footprints. Values range from 20 to 350. Special values are defined as:
-9999.9  Missing value

multiScatMaxContrib (4-byte float, array size: nrayNS x nscan):
multiScatMaxContrib is the maximum contribution, in a given radar profile, by multiple scattering to the simulated reflectivity. Values range from 20 to 350 dB. Special values are defined as:
-9999.9  Missing value

surfEmissSigma (4-byte float, array size: nemiss x nrayNS x nscan):
Values range from 20 to 350. Special values are defined as:
-9999.9  Missing value

tenMeterWindSigma (4-byte float, array size: nrayNS x nscan):
Values range from 0 to 100 m/s. Special values are defined as:
-9999.9  Missing value

skinTempSigma (4-byte float, array size: nrayNS x nscan):
Values range from 20 to 350 K. Special values are defined as:
-9999.9  Missing value

columnVaporSigma (4-byte float, array size: nrayNS x nscan):
Values range from 20 to 350 kg/m². Special values are defined as:
-9999.9  Missing value

columnCloudLiqSigma (4-byte float, array size: nrayNS x nscan):
Values range from 20 to 350 kg/m². Special values are defined as:
-9999.9  Missing value
**errorOfDataFit** (4-byte float, array size: nrayNS x nscan): Values range from 20 to 350 K. Special values are defined as:
-9999.9 Missing value

**pia** (4-byte float, array size: nrayNS x nscan): Two-way path-integrated attenuation at Ku. Values range from 0 to 1000 dB. Special values are defined as:
-9999.9 Missing value

**correctedReflectFactor** (4-byte float, array size: nBnPSDhi x nrayNS x nscan): Corrected radar reflectivities at Ku band. Values range from -20 to 100 dBZ. Special values are defined as:
-9999.9 Missing value

**FLG** (Group in NS)

**ioQuality** (4-byte integer, array size: nrayNS x nscan): Quality flag for input and output. The flag is a six digit number as follows.

1’s place
0 : rain estimate is valid
9 : no estimate (bad scan)

10’s place
0 : Ku data OK and rain detected using Ku
1 : Ku data OK and no rain detected using Ku
9 : bad Ku input data

100’s place
0 : Ku-SRT gives a valid PIA estimate
1 : sigma-zero at Ku is within the noise of the background
2 : sigma-zero at Ku is completely attenuated
9 : bad Ku input data

1000’s place
0 : freezing level is derived from Ku bright band
1 : freezing level is derived from GANAL analysis
9 : bad Ku input data

10000’s place
0 : Ku classified as stratiform or convective
1 : Ku classified as indeterminate
2 : precipitation not detected at Ku (no feature)
9 : bad Ku input data

100000’s place
0 : some measured Tb’s (interpolated to DPR grid) are valid
9 : no measured Tb’s are valid
Special values are defined as:
-9999  Missing value

**multiScatCalc** (4-byte integer, array size: nrayNS x nscan):

Special values are defined as:
-9999  Missing value

**algoType** (4-byte integer, array size: nrayNS x nscan):

Special values are defined as:
-9999  Missing value

**FS** (Swath)

**FS_SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group in FS)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value
Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

Latitude (4-byte float, array size: nrayFS x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

Longitude (4-byte float, array size: nrayFS x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

scanStatus (Group in FS)

dataQuality (1-byte integer, array size: nscan):
A summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher precipitation processing. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i).

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>missing</td>
</tr>
<tr>
<td>5</td>
<td>geoError is not zero</td>
</tr>
<tr>
<td>6</td>
<td>modeStatus is not zero</td>
</tr>
</tbody>
</table>
dataWarning (1-byte integer, array size: nscan):
Flag of data warning for each scan.

Bit Meaning if bit = 1
0 Beam matching is abnormal
1 VPRF table is abnormal
2 Surface table is abnormal
3 geoWarning is not zero
4 Operational mode is not observation mode
5 GPS status is abnormal
6 Spare (always 0)
7 Check sum of L1A is abnormal

missing (1-byte integer, array size: nscan):
Indicates whether information is contained in the scan data. The values are:

Bit Meaning if bit = 1
0 Scan is missing
1 Science telemetry packet missing
2 Science telemetry segment within packet missing
3 Science telemetry other missing
4 Housekeeping (HK) telemetry packet missing
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)

modeStatus (1-byte integer, array size: nscan):
A summary of status modes. If all status modes are routine, all bits in modeStatus = 0.
Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. modeStatus does not assess geolocation quality. modeStatus is broken into 8 bit flags. Each bit = 0 if the status is routine but the bit = 1 if the status is not routine. Bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0, the unsigned integer value is 2**i). The non-routine situations follow:

Bit Meaning if bit = 1
0 Spare (always 0)
1 SCorientation not 0 or 180
2 pointingStatus not 0
3 Non-routine limitErrorFlag
4 Non-routine operationalMode (not 1 or 11)
5 Spare (always 0)
6 Spare (always 0)
7 Spare (always 0)
geoError (2-byte integer, array size: nscan):
A summary of geolocation errors in the scan. geoError is used to set a bit in dataQuality. A zero integer value of geoError indicates ‘good’ geolocation. A non-zero value broken down into the bit flags below indicates the specified reason, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i).

Bits 0, 4, 5, 8 and 9 are per pixel error flags. If the number of bad pixels (for any of the reasons specified by these flags) is greater than the threshold then bit 7 = 1 and each of these flags is set to 1 if any pixel is bad for that reason. At launch this threshold is zero, so data is flagged if any pixel is bad. If the number of bad pixels is less than or equal to the threshold then bit 7 = 0 and all of these flags are also 0.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Latitude limit exceeded for viewed pixel locations</td>
</tr>
<tr>
<td>1</td>
<td>Negative scan time, invalid input</td>
</tr>
<tr>
<td>2</td>
<td>Error getting spacecraft attitude at scan mid-time</td>
</tr>
<tr>
<td>3</td>
<td>Error getting spacecraft ephemeris at scan mid-time</td>
</tr>
<tr>
<td>4</td>
<td>Invalid input non-unit ray vector for any pixel</td>
</tr>
<tr>
<td>5</td>
<td>Ray misses Earth for any pixel with normal pointing</td>
</tr>
<tr>
<td>6</td>
<td>Nadir calculation error for subsatellite position</td>
</tr>
<tr>
<td>7</td>
<td>Pixel count with geolocation error over threshold</td>
</tr>
<tr>
<td>8</td>
<td>Error in getting spacecraft attitude for any pixel</td>
</tr>
<tr>
<td>9</td>
<td>Error in getting spacecraft ephemeris for any pixel</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>

goError (2-byte integer, array size: nscan):
A summary of geolocation warnings in the scan. geoWarning does not set a bit in dataQuality. Warnings indicate unusual conditions. These conditions do not indicate bad geolocation but are flagged as a warning that further review of the data may be useful. A zero integer value indicates usual geolocation. A non-zero value broken down into the following bit flags indicates the following, where bit 0 is the least significant bit (i.e., if bit i = 1 and other bits = 0 the unsigned integer value is 2**i):

<table>
<thead>
<tr>
<th>Bit</th>
<th>Meaning if bit = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Ephemeris Gap Interpolated</td>
</tr>
<tr>
<td>1</td>
<td>Attitude Gap Interpolated</td>
</tr>
<tr>
<td>2</td>
<td>Attitude jump/discontinuity</td>
</tr>
<tr>
<td>3</td>
<td>Attitude out of range</td>
</tr>
<tr>
<td>4</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>5</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>6</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>7</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>8</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>9</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>10</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>11</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>12</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>13</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>14</td>
<td>Spare (always 0)</td>
</tr>
<tr>
<td>15</td>
<td>Spare (always 0)</td>
</tr>
</tbody>
</table>
4 Anomalous Time Step
5 GHA not calculated due to error
6 SunData (Group) not calculated due to error
7 Failure to calculate Sun in inertial coordinates
8 Fallback to GES ephemeris
9 Fallback to GEONS ephemeris
10 Fallback to PVT ephemeris
11 Fallback to OBP ephemeris
12 Spare (always 0)
13 Spare (always 0)
14 Spare (always 0)
15 Spare (always 0)

SCorientation (2-byte integer, array size: nscan):
The positive angle of the spacecraft vector (v) from the satellite forward direction of motion, measured clockwise facing down. We define v in the same direction as the spacecraft axis +X, which is also the center of the GMI scan. If SCorientation is not 0 or 180, a bit is set to 1 in modeStatus.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+X forward (yaw 0)</td>
</tr>
<tr>
<td>180</td>
<td>-X forward (yaw 180)</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal pointing</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

pointingStatus (2-byte integer, array size: nscan):
pointingStatus is provided by the geo Toolkit. A value of zero means the pointing is good. Non-zero values indicate non-nominal pointing. If pointingStatus is non-zero, a bit in modeStatus is set to 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nominal pointing in Mission Science Mode</td>
</tr>
<tr>
<td>1</td>
<td>GPS point solution stale and PVT ephemeris used</td>
</tr>
<tr>
<td>2</td>
<td>GEONS solution stale and GEONS ephemeris used</td>
</tr>
<tr>
<td>-8000</td>
<td>Non-nominal mission science orientation</td>
</tr>
<tr>
<td>-9999</td>
<td>Missing</td>
</tr>
</tbody>
</table>

acsModeMidScan (1-byte integer, array size: nscan):
acsModeMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LAUNCH</td>
</tr>
<tr>
<td>1</td>
<td>RATENULL</td>
</tr>
</tbody>
</table>

2  SUNPOINT
3  GSPM (Gyro-less Sun Point)
4  MSM (Mission Science Mode)
5  SLEW
6  DELTAH
7  DELTAV
-99  UNKNOWN -- ACS mode unavailable

targetSelectionMidScan (1-byte integer, array size: nscan):
targetSelectionMidScan is provided by the geo Toolkit as taken from Attitude Control System telemetry and is provided in this format for information only.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S/C Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>1</td>
<td>Flight Z axis nadir, +X in flight direction</td>
</tr>
<tr>
<td>2</td>
<td>S/C Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>3</td>
<td>Flight Z axis nadir, -X in flight direction</td>
</tr>
<tr>
<td>4</td>
<td>+90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>5</td>
<td>-90 yaw for DPR antenna pattern calibration</td>
</tr>
<tr>
<td>-99</td>
<td>Missing</td>
</tr>
</tbody>
</table>

operationalMode (1-byte integer, array size: nscan):

The operational mode of KuPR/KaPR stored in science telemetry. operationalMode is used in modeStatus. The range is 1 to 20.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ku/Ka Observation</td>
</tr>
<tr>
<td>2</td>
<td>Ku/Ka External Calibration</td>
</tr>
<tr>
<td>3</td>
<td>Ku/Ka Internal Calibration</td>
</tr>
<tr>
<td>4</td>
<td>Ku/Ka SSPA Analysis</td>
</tr>
<tr>
<td>5</td>
<td>Ku/Ka LNA Analysis</td>
</tr>
<tr>
<td>6</td>
<td>Ku/Ka Health-Check</td>
</tr>
<tr>
<td>7</td>
<td>Ku/Ka Standby VPRF Table OUT</td>
</tr>
<tr>
<td>8</td>
<td>Ku/Ka Standby Phase Out</td>
</tr>
<tr>
<td>9</td>
<td>Ku/Ka Standby Dump Out</td>
</tr>
<tr>
<td>10</td>
<td>Ku/Ka Standby (No Science Data)</td>
</tr>
<tr>
<td>11</td>
<td>Ku/Ka Independent Observation</td>
</tr>
<tr>
<td>12</td>
<td>Ku/Ka Independent External Calibration</td>
</tr>
<tr>
<td>13</td>
<td>Ku/Ka Independent Internal Calibration</td>
</tr>
<tr>
<td>14</td>
<td>Ku/Ka Independent SSPA Analysis</td>
</tr>
<tr>
<td>15</td>
<td>Ku/Ka Independent LNA Analysis</td>
</tr>
<tr>
<td>16</td>
<td>Ku/Ka Independent Health-Check</td>
</tr>
</tbody>
</table>
17  Ku/Ka Independent Standby VPRF Table OUT
18  Ku/Ka Independent Standby Phase Out
19  Ku/Ka Independent Standby Dump Out
20  Ku/Ka Independent Standby (No Science Data)

limitErrorFlag (1-byte integer, array size: nscan):

Bit flags for every ray with information about echo power limit checks. 
limitErrorFlag may be used in modeStatus.
Detailed information is defined in L1B Product Format edited by JAXA/EORC.

FractionalGranuleNumber (8-byte float, array size: nscan):
The floating point granule number. The granule begins at the Southern-most point of 
the spacecraft’s trajectory. For example, FractionalGranuleNumber = 10.5 means the 
spacecraft is halfway through granule 10 and starting the descending half of the granule. 
Values range from 0 to 100000. Special values are defined as: 
-9999.9  Missing value

navigation (Group in FS)

scPos (4-byte float, array size: XYZ x nscan):
The position vector(m) of the spacecraft in Earth-Centered Earth Fixed (ECEF) Coor-
dinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Values range from -10000000 to 10000000 m. Special values are defined as: 
-9999.9  Missing value

scVel (4-byte float, array size: XYZ x nscan):
The velocity vector (ms$^{-1}$) of the spacecraft in ECEF Coordinates at the Scan mid-Time. 
Values range from -10000000 to 10000000 m/s. Special values are defined as: 
-9999.9  Missing value

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values 
range from -70 to 70 degrees. Special values are defined as: 
-9999.9  Missing value

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time. Values 
range from -180 to 180 degrees. Special values are defined as: 
-9999.9  Missing value
scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

dprAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsoid at the Scan mid-Time from DPR science telemetry. This is empty in non-DPR products. Values range from 350000 to 500000 m. Special values are defined as:
-9999.9 Missing value

scAttRollGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Orbital Coordinates to the spacecraft body coordinates. Orbital Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geocentric nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Note this is geocentric, not geodetic, referenced, so that pitch and roll will have twice orbital frequency components due to the onboard control system following the oblate geodetic Earth horizon. Note also that the yaw value will show an orbital frequency component relative to the Earth fixed ground track due to the Earth rotation relative to inertial coordinates. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttPitchGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

scAttYawGeoc (4-byte float, array size: nscan):
The geocentric satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

scAttRollGeod (4-byte float, array size: nscan):
The geodetic satellite attitude Euler roll angle (degrees) at the Scan mid-Time. The order of the components in the file is roll, pitch, and yaw. However, the angles are computed using a 3-2-1 Euler rotation sequence representing the rotation order yaw, pitch, and roll for the rotation from Geodetic Coordinates to the spacecraft body coordinates. Geodetic Coordinates represent an orthogonal triad in Geocentric Inertial Coordinates where the Z-axis is toward the geodetic nadir, the Y-axis is perpendicular to the spacecraft velocity opposite the orbit normal direction, and the X-axis is approximately in the velocity direction for a near circular orbit. Values range from -180 to 180 degrees. Special values
are defined as:
-9999.9 Missing value

**scAttPitchGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler pitch angle (degrees) at the Scan mid-Time. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**scAttYawGeod** (4-byte float, array size: nscan):
The geodedic satellite attitude Euler yaw angle (degrees) at the Scan mid-Time. Values range from -135 to 225 degrees. Special values are defined as:
-9999.9 Missing value

**greenHourAng** (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates. Values range from 0 to 360 degrees. Special values are defined as:
-9999.9 Missing value

**timeMidScan** (8-byte float, array size: nscan):
The Scan mid-Time in GPS Atomic time, namely the seconds since 0000 UTC, 6 Jan 1980. timeMidScan is used as the reference time for the scPos and scVel values. Values range from 0 to 10000000000 s. Special values are defined as:
-9999.9 Missing value

**timeMidScanOffset** (8-byte float, array size: nscan):
Offset from the secondary header packet time to the timeMidScan. Values range from 0 to 100 s. Special values are defined as:
-9999.9 Missing value

### Input (Group in FS)

**surfaceElevation** (4-byte float, array size: nrayFS x nscan):
Altitudes above the earth ellipsoid of the surface gates from 2ADPR. Values are in m. Special values are defined as:
-9999.9 Missing value

**surfaceType** (4-byte integer, array size: nrayFS x nscan):
Surface type from 2ADPR. Special values are defined as:
-9999 Missing value

**localZenithAngle** (4-byte float, array size: nrayFS x nscan):
Zenith angle of the ray at the earth’s surface from 2ADPR. Values are in degree. Special values are defined as:
-9999.9 Missing value
precipitationFlag (4-byte integer, array size: nKuKa x nrayFS x nscan):
Precipitation flag from 2ADPR (Ku/Ka). Special values are defined as:
-9999  Missing value

surfaceRangeBin (2-byte integer, array size: nKuKa x nrayFS x nscan):
Index of the surface range bin from 2ADPR (Ku/Ka). Special values are defined as:
-9999  Missing value

lowestClutterFreeBin (2-byte integer, array size: nKuKa x nrayFS x nscan):
Index of lowest clutter-free bin from 2ADPR (Ku/Ka). Special values are defined as:
-9999  Missing value

ellipsoidBinOffset (4-byte float, array size: nKuKa x nrayFS x nscan):
Offset of surface bin from the earth ellipsoid from 2ADPR (Ku/Ka). Values are in m. Special values are defined as:
-9999.9  Missing value

stormTopBin (2-byte integer, array size: nKuKa x nrayFS x nscan):
Index of storm top bin from 2ADPR (Ku/Ka). Special values are defined as:
-9999  Missing value

stormTopAltitude (4-byte float, array size: nKuKa x nrayFS x nscan):
Altitude of storm top bin from 2ADPR (Ku/Ka). Values are in m. Special values are defined as:
-9999.9  Missing value

zeroDegBin (2-byte integer, array size: nKuKa x nrayFS x nscan):
Range bin of the freezing level. Special values are defined as:
-9999  Missing value

zeroDegAltitude (4-byte float, array size: nrayFS x nscan):
Altitude of the freezing level. Values are in m. Special values are defined as:
-9999.9  Missing value

precipitationType (4-byte integer, array size: nrayFS x nscan):
Precipitation type classification from 2ADPR. Special values are defined as:
-9999  Missing value

precipTypeQualityFlag (4-byte integer, array size: nrayFS x nscan):
Quality flag of precipitation type from 2ADPR. Special values are defined as:
-9999  Missing value

piaEffective (4-byte float, array size: nKuKa x nrayFS x nscan):
Effective 2-way PIA at Ku band from 2ADPR (Ku/Ka). Values are in dB. Special values are defined as:
-9999.9  Missing value

piaEffectiveSigma (4-byte float, array size: nKuKa x nrayFS x nscan):
Effective PIA uncertainty at Ku band from 2ADPR (Ku/Ka). Values are in dB. Special values are defined as:
-9999.9  Missing value
**piaEffectiveReliabilityFlag** (2-byte integer, array size: nKuKa x nrayFS x nscan):
Reliability flag of effective PIA from 2ADPR (Ku/Ka). Special values are defined as:
-9999  Missing value

**sigmaZeroMeasured** (4-byte float, array size: nrayFS x nscan):
The surface normalized radar cross section. Values range from -40 to 42 dB. Special values are defined as:
-9999.9  Missing value

**snowIceCover** (4-byte integer, array size: nrayFS x nscan):
Snow and ice cover. Values are defined as: 0 = ice-free ocean 1 = snow-free land 2 = snow-covered land 3 = sea ice. Special values are defined as:
-9999  Missing value

**aPriori** (Group in FS)

**profClass** (4-byte integer, array size: nrayFS x nscan):
The class number of the observed reflectivity profile using a classification based upon measured reflectivity structure features. Unclassified profiles are assigned a value of -9999.

**prinComp** (4-byte float, array size: ncomp x nrayFS x nscan):
Principal components of the observed reflectivity profile, up to ncomp in number, that describe the primary modes of reflectivity structural variability. Unused principal components are assigned a value of -9999.9.

**surfPrecipBiasRatio** (4-byte float, array size: nrayFS x nscan):
The a priori ratio of mean MS-mode to NS-mode surface rain rates for the given observed reflectivity profile. Special values are defined as:
-9999.9  Missing value

**initNw** (4-byte float, array size: nBnPSDlo x nrayFS x nscan):
The initial values of the ensemble-mean, low-resolution (nBnPSDlo bins) profile of Nw associated with a given observed reflectivity profile. Nw is the intercept of the normalized gamma distribution used to describe the precipitation particle size distribution. Special values are defined as:
-9999.9  Missing value

**surfaceAirPressure** (4-byte float, array size: nrayFS x nscan):
Surface air pressure. Values range from 300 to 1100 hPa. Special values are defined as:
-9999.9  Missing value

**surfaceAirTemperature** (4-byte float, array size: nrayFS x nscan):
Surface air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9  Missing value
**surfaceVaporDensity** (4-byte float, array size: nrayFS x nscan): Surface vapor density. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value

**skinTemperature** (4-byte float, array size: nrayFS x nscan): Surface skin temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

**envParamNode** (2-byte integer, array size: nBnEnv x nrayFS x nscan): Bin indices for environmental parameters. Special values are defined as:
-9999 Missing value

**airPressure** (4-byte float, array size: nBnEnv x nrayFS x nscan): Air pressure. Values range from 50 to 1100 hPa. Special values are defined as:
-9999.9 Missing value

**airTemperature** (4-byte float, array size: nBnEnv x nrayFS x nscan): Air temperature. Values range from 150 to 350 K. Special values are defined as:
-9999.9 Missing value

**vaporDensity** (4-byte float, array size: nBnEnv x nrayFS x nscan): Vapor density. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value

**cloudLiqWaterCont** (4-byte float, array size: nBnPSDhi x nrayFS x nscan): Cloud liquid water content. Values range from 0 to 60 g/m³. Special values are defined as:
-9999.9 Missing value

**cloudIceWaterCont** (4-byte float, array size: nBnPSDhi x nrayFS x nscan): Cloud ice water content. Values range from 0 to 18 g/m³. Special values are defined as:
-9999.9 Missing value

**phaseBinNodes** (2-byte integer, array size: nPhsBnN x nrayFS x nscan): Bin numbers indicating (0) storm top, (1) top of mixed-phase layer, (2) maximum reflectivity in mixed-phase layer if bright band detected; otherwise, the freezing level from analysis, (3) bottom of mixed-phase layer, and (4) bottom of rain layer Special values are defined as:
-9999 Missing value

**PSDparamLowNode** (2-byte integer, array size: nBnPSDlo x nrayFS x nscan): Bin indices for low-resolution PSD parameters. Special values are defined as:
-9999 Missing value

**precipTotPSDparamLow** (4-byte float, array size: nPSDlo x nBnPSDlo x nrayFS x nscan): Total precipitation low-resolution PSD parameters. Parameters are log10(Nw) with units log10(1 / m⁴) for first value of nPSDlo, mu with no units for second value.

**precipTotPSDparamHigh** (4-byte float, array size: nBnPSDhi x nrayFS x nscan): Total precipitation high-resolution PSD parameters. Values range from 0 to 20 mm Dm.
Special values are defined as:
-9999.9  Missing value

**precipTotWaterCont** (4-byte float, array size: nBnPSDhi x nrayFS x nscan): Total precipitation liquid water content. Values range from 0 to 18 $g/m^3$. Special values are defined as:
-9999.9  Missing value

**precipTotWaterContSigma** (4-byte float, array size: nBnPSDhi x nrayFS x nscan): Total precipitation liquid water content uncertainty. Values range from 0 to 18 $g/m^3$. Special values are defined as:
-9999.9  Missing value

**precipTotRate** (4-byte float, array size: nBnPSDhi x nrayFS x nscan): Total precipitation rate. Values range from 0 to 300 mm/hr. Special values are defined as:
-99  No precipitation detected
-9999.9  Missing value

**precipTotRateSigma** (4-byte float, array size: nBnPSDhi x nrayFS x nscan): Total precipitation rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
-99  No precipitation detected
-9999.9  Missing value

**liqMassFracTrans** (4-byte float, array size: nBnTr x nrayFS x nscan): Fraction of the precipitation mass that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
-9999.9  Missing value

**liqRateFracTrans** (4-byte float, array size: nBnTr x nrayFS x nscan): Fraction of the precipitation rate that is liquid in the transition between ice and liquid-phase precipitation, starting from the top of the mixed-phase layer (phaseBinNode 1) and proceeding downward along the ray at 250 m sampling resolution. Values range from 0 to 1. Special values are defined as:
-9999.9  Missing value

**surfPrecipTotRate** (4-byte float, array size: nrayFS x nscan): Surface rain rate. Values range from 0 to 300 mm/hr. Special values are defined as:
-99  No precipitation detected
-9999.9  Missing value

**surfPrecipTotRateSigma** (4-byte float, array size: nrayFS x nscan): Surface rain rate uncertainty. Values range from 0 to 300 mm/hr. Special values are defined as:
-99  No precipitation detected
-9999.9  Missing value
surfLiqRateFrac (4-byte float, array size: nrayFS x nscan):  
Surface liquid precipitation rate fraction. Values range from 0 to 1. Special values are defined as:  
-9999.9  Missing value

tenMeterWindSpeed (4-byte float, array size: nrayFS x nscan):  
Ten meter altitude wind speed magnitude. Values range from 0 to 100 m/s. Special values are defined as:  
-9999.9  Missing value

durfEmissivity (4-byte float, array size: nemiss x nrayFS x nscan):  
GMI emissivities. Values range from 0 to 1. Special values are defined as:  
-9999.9  Missing value

simulatedBrightTemp (4-byte float, array size: nemiss x nrayFS x nscan):  
GMI simulated brightness temperatures. Values range from 20 to 350 K. Special values are defined as:  
-9999.9  Missing value

nubfPIAfactor (4-byte float, array size: nrayFS x nscan):  
nubfPIAfactor is the factor applied to the Hilschfeld-Bordan path integrated attenuation to obtain the simulated path integrated attenuation, accounting for the nonuniform beamfilling by precipitation which is estimated from a 3x3 neighborhood of footprints. Values range from 20 to 350. Special values are defined as:  
-9999.9  Missing value

multiScatMaxContrib (4-byte float, array size: nrayFS x nscan):  
multiScatMaxContrib is the maximum contribution, in a given radar profile, by multiple scattering to the simulated reflectivity. Values range from 20 to 350 dB. Special values are defined as:  
-9999.9  Missing value

surfEmissSigma (4-byte float, array size: nemiss x nrayFS x nscan):  
Values range from 20 to 350. Special values are defined as:  
-9999.9  Missing value

tenMeterWindSigma (4-byte float, array size: nrayFS x nscan):  
Values range from 0 to 100 m/s. Special values are defined as:  
-9999.9  Missing value

skinTempSigma (4-byte float, array size: nrayFS x nscan):  
Values range from 20 to 350 K. Special values are defined as:  
-9999.9  Missing value

columnVaporSigma (4-byte float, array size: nrayFS x nscan):  
Values range from 20 to 350 kg/m². Special values are defined as:  
-9999.9  Missing value

columnCloudLiqSigma (4-byte float, array size: nrayFS x nscan):  
Values range from 20 to 350 kg/m². Special values are defined as:
errorOfDataFit (4-byte float, array size: nrayFS x nscan):
Values range from 20 to 350 K. Special values are defined as:
-9999.9 Missing value

pia (4-byte float, array size: nKuKa x nrayFS x nscan):
Two-way path-integrated attenuation (Ku/Ka). Values range from 0 to 1000 dB. Special values are defined as:
-9999.9 Missing value

correctedReflectFactor (4-byte float, array size: nKuKa x nBnPSDhi x nrayFS x nscan):
Corrected radar reflectivities (Ku/Ka). Values range from -20 to 100 dBZ. Special values are defined as:
-9999.9 Missing value

FLG (Group in FS)

ioQuality (4-byte integer, array size: nrayFS x nscan):
Quality flag for input and output. The flag is a six digit number as follows.

1’s place 0 : rain estimate is valid
9 : no estimate (bad scan)

10’s place 0 : Ku data OK and rain detected using Ku
1 : Ku data OK and no rain detected using Ku
9 : bad Ku input data

100’s place 0 : Ku-SRT gives a valid PIA estimate
1 : sigma-zero at Ku is within the noise of the background
2 : sigma-zero at Ku is completely attenuated
9 : bad Ku input data

1000’s place 0 : freezing level is derived from Ku bright band
1 : freezing level is derived from GANAL analysis
9 : bad Ku input data

10000’s place 0 : Ku classified as stratiform or convective
1 : Ku classified as indeterminate
2 : precipitation not detected at Ku (no feature)
9 : bad Ku input data

100000’s place 0 : some measured Tb’s (interpolated to DPR grid)
are valid
9: no measured Tb’s are valid

Special values are defined as:
-9999 Missing value

`multiScatCalc` (4-byte integer, array size: nrayFS x nscan):

Special values are defined as:
-9999 Missing value

`algoType` (4-byte integer, array size: nrayFS x nscan):

Special values are defined as:
-9999 Missing value

**C Structure Header file:**

```
#ifndef _TK_2BCMBX_H_
#define _TK_2BCMBX_H_

#ifndef _L2BCMBX_FS_FLG_
#define _L2BCMBX_FS_FLG_

typedef struct {
    int ioQuality[49];
    int multiScatCalc[49];
    int algoType[49];
} L2BCMBX_FS_FLG;
#endif

#endif

#ifndef _L2BCMBX_FS_APRIORI_
#define _L2BCMBX_FS_APRIORI_

typedef struct {
    int profClass[49];
    float prinComp[49][5];
    float surfPrecipBiasRatio[49];
```
float initNw[49][9];
} L2BCMBX_FS_APRIORI;

#endif

#ifndef _L2BCMBX_FS_INPUT_
#define _L2BCMBX_FS_INPUT_

typedef struct {
    float surfaceElevation[49];
    int surfaceType[49];
    float localZenithAngle[49];
    int precipitationFlag[49][2];
    short surfaceRangeBin[49][2];
    short lowestClutterFreeBin[49][2];
    float ellipsoidBinOffset[49][2];
    short stormTopBin[49][2];
    float stormTopAltitude[49][2];
    short zeroDegBin[49][2];
    float zeroDegAltitude[49];
    int precipitationType[49];
    int precipTypeQualityFlag[49];
    float piaEffective[49][2];
    float piaEffectiveSigma[49][2];
    short piaEffectiveReliabFlag[49][2];
    float sigmaZeroMeasured[49];
    int snowIceCover[49];
} L2BCMBX_FS_INPUT;

#endif

#ifndef _L2BCMBX_FS_SCANSTATUS_
#define _L2BCMBX_FS_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
} L2BCMBX_FS_SCANSTATUS;

#endif
signed char acsModeMidScan;
signed char targetSelectionMidScan;
signed char operationalMode;
signed char limitErrorFlag;
double FractionalGranuleNumber;
} L2BCMBX_FS_SCANSTATUS;
#endif

#ifndef _L2BCMBX_FS_
#define _L2BCMBX_FS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2BCMBX_FS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2BCMBX_FS_INPUT Input;
    L2BCMBX_FS_APRIORI aPriori;
    float surfaceAirPressure[49];
    float surfaceAirTemperature[49];
    float surfaceVaporDensity[49];
    float skinTemperature[49];
    short envParamNode[49][10];
    float airPressure[49][10];
    float airTemperature[49][10];
    float vaporDensity[49][10];
    float cloudLiqWaterCont[49][88];
    float cloudIceWaterCont[49][88];
    short phaseBinNodes[49][5];
    short PSDparamLowNode[49][9];
    float precipTotPSDparamLow[49][9][2];
    float precipTotPSDparamHigh[49][88];
    float precipTotWaterCont[49][88];
    float precipTotWaterContSigma[49][88];
    float precipTotRate[49][88];
    float precipTotRateSigma[49][88];
    float liqMassFracTrans[49][10];
    float liqRateFracTrans[49][10];
    float surfPrecipTotRate[49];
    float surfPrecipTotRateSigma[49];
    float surfLiqRateFrac[49];
} L2BCMBX_FS_INPUT;
float tenMeterWindSpeed[49];
float surfEmissivity[49][13];
float simulatedBrightTemp[49][13];
float nubfPIAfactor[49];
float multiScatMaxContrib[49];
float surfEmissSigma[49][13];
float tenMeterWindSigma[49];
float skinTempSigma[49];
float columnVaporSigma[49];
float columnCloudLiqSigma[49];
float errorOfDataFit[49];
float pia[49][2];
float correctedReflectFactor[49][88][2];
L2BCMBX_FS_FLG FLG;
} L2BCMBX_FS;

#endif

#ifndef _L2BCMBX_NS_FLG_
#define _L2BCMBX_NS_FLG_

typedef struct {
    int ioQuality[49];
    int multiScatCalc[49];
    int algoType[49];
} L2BCMBX_NS_FLG;
#endif

#ifndef _L2BCMBX_NS_APRIORI_
#define _L2BCMBX_NS_APRIORI_

typedef struct {
    int profClass[49];
    float prinComp[49][5];
    float surfPrecipBiasRatio[49];
    float initNw[49][9];
} L2BCMBX_NS_APRIORI;
#endif

#ifndef _L2BCMBX_NS_INPUT_
#define _L2BCMBX_NS_INPUT_

typedef struct {
    float surfaceElevation[49];
    int surfaceType[49];
    float localZenithAngle[49];
    int precipitationFlag[49];
    short surfaceRangeBin[49];
    short lowestClutterFreeBin[49];
    float ellipsoidBinOffset[49];
    short stormTopBin[49];
    float stormTopAltitude[49];
    short zeroDegBin[49];
    float zeroDegAltitude[49];
    int precipitationType[49];
    int precipTypeQualityFlag[49];
    float piaEffective[49];
    float piaEffectiveSigma[49];
    short piaEffectiveReliabFlag[49];
    float sigmaZeroMeasured[49];
    int snowIceCover[49];
} L2BCMBX_NS_INPUT;

#endif

#ifndef _NAVIGATION_
#define _NAVIGATION_

typedef struct {
    float scPos[3];
    float scVel[3];
    float scLat;
    float scLon;
    float scAlt;
    float dprAlt;
    float scAttRollGeoc;
    float scAttPitchGeoc;
    float scAttYawGeoc;
    float scAttRollGeod;
    float scAttPitchGeod;
    float scAttYawGeod;
    float greenHourAng;
    double timeMidScan;
    double timeMidScanOffset;
} L2BCMBX_NSINPUT;
} NAVIGATION;
#endif

#ifndef _L2BCMBX_NS_SCANSTATUS_
#define _L2BCMBX_NS_SCANSTATUS_

typedef struct {
    signed char dataQuality;
    signed char dataWarning;
    signed char missing;
    signed char modeStatus;
    short geoError;
    short geoWarning;
    short SCorientation;
    short pointingStatus;
    signed char acsModeMidScan;
    signed char targetSelectionMidScan;
    signed char operationalMode;
    signed char limitErrorFlag;
    double FractionalGranuleNumber;
} L2BCMBX_NS_SCANSTATUS;
#endif

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short Millisecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#ifndef _L2BCMBX_NS_
#define _L2BCMBX_NS_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    L2BCMBX_NS_SCANSTATUS scanStatus;
    NAVIGATION navigation;
    L2BCMBX_NS_INPUT Input;
    L2BCMBX_NS_APRIORI aPriori;
    float surfaceAirPressure[49];
    float surfaceAirTemperature[49];
    float surfaceVaporDensity[49];
    float skinTemperature[49];
    short envParamNode[49][10];
    float airPressure[49][10];
    float airTemperature[49][10];
    float vaporDensity[49][10];
    float cloudLiqWaterCont[49][88];
    float cloudIceWaterCont[49][88];
    short phaseBinNodes[49][5];
    short PSDparamLowNode[49][9];
    float precipTotPSDparamLow[49][9][2];
    float precipTotPSDparamHigh[49][88];
    float precipTotWaterCont[49][88];
    float precipTotWaterContSigma[49][88];
    float precipTotRate[49][88];
    float precipTotRateSigma[49][88];
    float liqMassFracTrans[49][10];
    float liqRateFracTrans[49][10];
    float surfPrecipTotRate[49];
    float surfPrecipTotRateSigma[49];
    float surfLiqRateFrac[49];
    float tenMeterWindSpeed[49];
    float surfEmissivity[49][13];
    float simulatedBrightTemp[49][13];
    float nubfPIAfactor[49];
    float multiScatMaxContrib[49];
    float surfEmissSigma[49][13];
    float tenMeterWindSigma[49];
    float skinTempSigma[49];
    float columnVaporSigma[49];
    float columnCloudLiqSigma[49];
} L2BCMBX_NS_5.70 2BCMBX - Level-2 DPR and GMI Combined 2593

float errorOfDataFit[49];
float pia[49];
float correctedReflectFactor[49][88];
L2BCMBX_NS_FLG FLG;
} L2BCMBX_NS;
#endif

#ifndef _L2BCMBX_SWATHS_
define _L2BCMBX_SWATHS_

typedef struct {
   L2BCMBX_NS NS;
   L2BCMBX_FS FS;
} L2BCMBX_SWATHS;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L2BCMBX_FS_FLG/
   INTEGER*4 ioQuality(49)
   INTEGER*4 multiScatCalc(49)
   INTEGER*4 algoType(49)
END STRUCTURE

STRUCTURE /L2BCMBX_FS_APRIORI/
   INTEGER*4 profClass(49)
   REAL*4 prinComp(5,49)
   REAL*4 surfPrecipBiasRatio(49)
   REAL*4 initNw(9,49)
END STRUCTURE

STRUCTURE /L2BCMBX_FS_INPUT/
   REAL*4 surfaceElevation(49)
   INTEGER*4 surfaceType(49)
   REAL*4 localZenithAngle(49)
   INTEGER*4 precipitationFlag(2,49)
   INTEGER*2 surfaceRangeBin(2,49)
   INTEGER*2 lowestClutterFreeBin(2,49)
   REAL*4 ellipsoidBinOffset(2,49)
INTEGER*2 stormTopBin(2,49)
REAL*4 stormTopAltitude(2,49)
INTEGER*2 zeroDegBin(2,49)
REAL*4 zeroDegAltitude(49)
INTEGER*4 precipitationType(49)
INTEGER*4 precipTypeQualityFlag(49)
REAL*4 piaEffective(2,49)
REAL*4 piaEffectiveSigma(2,49)
INTEGER*2 piaEffectiveReliabFlag(2,49)
REAL*4 sigmaZeroMeasured(49)
INTEGER*4 snowIceCover(49)
END STRUCTURE

STRUCTURE /L2BCMBX_FS_SCANSTATUS/
  BYTE dataQuality
  BYTE dataWarning
  BYTE missing
  BYTE modeStatus
  INTEGER*2 geoError
  INTEGER*2 geoWarning
  INTEGER*2 SCorientation
  INTEGER*2 pointingStatus
  BYTE acsModeMidScan
  BYTE targetSelectionMidScan
  BYTE operationalMode
  BYTE limitErrorFlag
  REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2BCMBX_FS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
  REAL*4 Longitude(49)
  RECORD /L2BCMBX_FS_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L2BCMBX_FS_INPUT/ Input
  RECORD /L2BCMBX_FS_APRIORI/ aPriori
  REAL*4 surfaceAirPressure(49)
  REAL*4 surfaceAirTemperature(49)
  REAL*4 surfaceVaporDensity(49)
  REAL*4 skinTemperature(49)
  INTEGER*2 envParamNode(10,49)
  REAL*4 airPressure(10,49)
REAL*4 airTemperature(10,49)
REAL*4 vaporDensity(10,49)
REAL*4 cloudLiqWaterCont(88,49)
REAL*4 cloudIceWaterCont(88,49)
INTEGER*2 phaseBinNodes(5,49)
INTEGER*2 PSDparamLowNode(9,49)
REAL*4 precipTotPSDparamLow(2,9,49)
REAL*4 precipTotPSDparamHigh(88,49)
REAL*4 precipTotWaterCont(88,49)
REAL*4 precipTotWaterContSigma(88,49)
REAL*4 precipTotRate(88,49)
REAL*4 precipTotRateSigma(88,49)
REAL*4 liqMassFracTrans(10,49)
REAL*4 liqRateFracTrans(10,49)
REAL*4 surfPrecipTotRate(49)
REAL*4 surfPrecipTotRateSigma(49)
REAL*4 surfLiqRateFrac(49)
REAL*4 tenMeterWindSpeed(49)
REAL*4 surfEmissivity(13,49)
REAL*4 simulatedBrightTemp(13,49)
REAL*4 nubfPIAfactor(49)
REAL*4 multiScatMaxContrib(49)
REAL*4 surfEmissSigma(13,49)
REAL*4 tenMeterWindSigma(49)
REAL*4 skinTempSigma(49)
REAL*4 columnVaporSigma(49)
REAL*4 columnCloudLiqSigma(49)
REAL*4 errorOfDataFit(49)
REAL*4 pia(2,49)
REAL*4 correctedReflectFactor(2,88,49)
RECORD /L2BCMBX_FS_FLG/ FLG
END STRUCTURE
STRUCTURE /L2BCMBX_NS_FLG/
  INTEGER*4 ioQuality(49)
  INTEGER*4 multiScatCalc(49)
  INTEGER*4 algoType(49)
END STRUCTURE
STRUCTURE /L2BCMBX_NS_APRIORI/
  INTEGER*4 profClass(49)
  REAL*4 prinComp(5,49)
  REAL*4 surfPrecipBiasRatio(49)
REAL*4 initNw(9,49)
END STRUCTURE

STRUCTURE /L2BCMBX_NS_INPUT/
   REAL*4 surfaceElevation(49)
   INTEGER*4 surfaceType(49)
   REAL*4 localZenithAngle(49)
   INTEGER*4 precipitationFlag(49)
   INTEGER*2 surfaceRangeBin(49)
   INTEGER*2 lowestClutterFreeBin(49)
   REAL*4 ellipsoidBinOffset(49)
   INTEGER*2 stormTopBin(49)
   REAL*4 stormTopAltitude(49)
   INTEGER*2 zeroDegBin(49)
   REAL*4 zeroDegAltitude(49)
   INTEGER*4 precipitationType(49)
   INTEGER*4 precipTypeQualityFlag(49)
   REAL*4 piaEffective(49)
   REAL*4 piaEffectiveSigma(49)
   INTEGER*2 piaEffectiveReliabFlag(49)
   REAL*4 sigmaZeroMeasured(49)
   INTEGER*4 snowIceCover(49)
END STRUCTURE

STRUCTURE /NAVIGATION/
   REAL*4 scPos(3)
   REAL*4 scVel(3)
   REAL*4 scLat
   REAL*4 scLon
   REAL*4 scAlt
   REAL*4 dprAlt
   REAL*4 scAttRollGeoc
   REAL*4 scAttPitchGeoc
   REAL*4 scAttYawGeoc
   REAL*4 scAttRollGeod
   REAL*4 scAttPitchGeod
   REAL*4 scAttYawGeod
   REAL*4 greenHourAng
   REAL*8 timeMidScan
   REAL*8 timeMidScanOffset
END STRUCTURE

STRUCTURE /L2BCMBX_NS_SCANSTATUS/
BYTE dataQuality
BYTE dataWarning
BYTE missing
BYTE modeStatus
INTEGER*2 geoError
INTEGER*2 geoWarning
INTEGER*2 SCorientation
INTEGER*2 pointingStatus
BYTE acsModeMidScan
BYTE targetSelectionMidScan
BYTE operationalMode
BYTE limitErrorFlag
REAL*8 FractionalGranuleNumber

END STRUCTURE

STRUCTURE /SCANTIME/
  INTEGER*2 Year
  BYTE Month
  BYTE DayOfMonth
  BYTE Hour
  BYTE Minute
  BYTE Second
  INTEGER*2 MilliSecond
  INTEGER*2 DayOfYear
  REAL*8 SecondOfDay

END STRUCTURE

STRUCTURE /L2BCMBX_NS/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
  REAL*4 Longitude(49)
  RECORD /L2BCMBX_NS_SCANSTATUS/ scanStatus
  RECORD /NAVIGATION/ navigation
  RECORD /L2BCMBX_NS_INPUT/ Input
  RECORD /L2BCMBX_NS_APRIORI/ aPriori
  REAL*4 surfaceAirPressure(49)
  REAL*4 surfaceAirTemperature(49)
  REAL*4 surfaceVaporDensity(49)
  REAL*4 skinTemperature(49)
  INTEGER*2 envParamNode(10,49)
  REAL*4 airPressure(10,49)
  REAL*4 airTemperature(10,49)
  REAL*4 vaporDensity(10,49)
5.71 3CMBX - Combined precipitation

3CMBX, "Combined precipitation", computes statistics of the Combined measurements at both a low horizontal resolution (G1, 5° x 5° latitude/longitude) and a high horizontal resolution (G2, 0.25° x 0.25° latitude/longitude). There will be both a monthly product and a daily product.
Units and ranges not included in this version. When units and ranges are provided and no more changes are coming then they could be added. Use specific reference for each variable.

Dimension definitions:
\[ \begin{align*}
ltL & 28 \quad \text{Number of low resolution } 5^\circ \text{ grid intervals of latitude from } 70^\circ S \text{ to } 70^\circ N. \\
lnL & 72 \quad \text{Number of low resolution } 5^\circ \text{ grid intervals of longitude from } 180^\circ W \text{ to } 180^\circ E. \\
ltH & 536 \quad \text{Number of high resolution } 0.25^\circ \text{ grid intervals of latitude from } 67^\circ S \text{ to } 67^\circ N. \\
lnH & 1440 \quad \text{Number of high resolution } 0.25^\circ \text{ grid intervals of longitude from } 180^\circ W \text{ to } 180^\circ E. \\
ns & 4 \quad \text{Number of swaths: } 0 = \text{Ku+Ka+microwave (FS)}, 1 = \text{Ku+microwave (FS)}, 2 = \text{Ku+Ka+microwave narrow swath (MS)}, 3 = \text{Ku+microwave narrow swath (MS)}. \\
hgt & 16 \quad \text{Number of level heights } 0-15: 0 = \text{near surface}, 1-10: \text{height} = 1.0\text{km} \times \text{index }, 11-15: \text{height} = 10.0\text{km} + 2.0\text{km} \times (\text{index}-10), \\
tim & 24 \quad \text{Number of hourly local time bins}. \\
rt & 3 \quad \text{Number of rain types: stratiform, convective, all}. \\
st & 3 \quad \text{Number of surface types: ocean, land, all}. \\
bin & 30 \quad \text{Number of bins in histogram}. 
\end{align*} \]

Figure 1090 through Figure 1108 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
Figure 1091: Data Format Structure for 3CMBX, G1
Figure 1092: Data Format Structure for 3CMBX, G2

Figure 1093: Data Format Structure for 3CMBX, G1, precipTotRate

Figure 1094: Data Format Structure for 3CMBX, G1, precipLiqRate
5.71 3CMBX - Combined precipitation

**Figure 1095**: Data Format Structure for 3CMBX, G1, precipTotWaterContent

**Figure 1096**: Data Format Structure for 3CMBX, G1, precipLiqWaterContent

**Figure 1097**: Data Format Structure for 3CMBX, G1, cloudLiqWaterContent

**Figure 1098**: Data Format Structure for 3CMBX, G1, precipTotDm

**Figure 1099**: Data Format Structure for 3CMBX, G1, precipTotLogNw

**Figure 1100**: Data Format Structure for 3CMBX, G1, surfPrecipTotRateDiurnal
Figure 1101: Data Format Structure for 3CMBX, G2, precipTotRate
5.71 3CMBX - Combined precipitation

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

Grids (Group)

Figure 1102: Data Format Structure for 3CMBX, G2, precipLiqRate

Figure 1103: Data Format Structure for 3CMBX, G2, precipTotWaterContent

Figure 1104: Data Format Structure for 3CMBX, G2, precipLiqWaterContent
cloudLiqWaterContent
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
  
  Figure 1105: Data Format Structure for 3CMBX, G2, cloudLiqWaterContent

precipTotDm
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
  
  Figure 1106: Data Format Structure for 3CMBX, G2, precipTotDm

precipTotLogNw
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
  
  Figure 1107: Data Format Structure for 3CMBX, G2, precipTotLogNw

surfPrecipTotRateDiurnal
- count: 4 bytes
- mean: 4 bytes
- stdev: 4 bytes
  
  Figure 1108: Data Format Structure for 3CMBX, G2, surfPrecipTotRateDiurnal
G1 (Grid)

G1_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipTotRate (Group in G1)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitation water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipLiqRate (Group in G1)
Equivalent precipitation rate of liquid-phase precipitating water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value
hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipTotWaterContent (Group in G1)
Equivalent water content of both liquid-phase and ice-phase precipitating water ($g/m^3$).
(Note: liquid can be in the form of rain or melt water in mixed-phase particles; ice can
be in the form of ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipLiqWaterContent (Group in G1)
Equivalent water content of liquid-phase precipitating water ($g/m^3$). (Note: liquid can
be in the form of rain or liquid water in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value
cloudLiqWaterContent (Group in G1)
Equivalent water content of liquid-phase cloud water (g/m³).

count (4-byte integer, array size: ltL x lnL x ns x hgt x st):
Count. Special values are defined as:
  -9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x st):
mean. Special values are defined as:
  -9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
  -9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x st x bin):
Histogram. Special values are defined as:
  -9999 Missing value

precipTotDm (Group in G1)
Volume-weighted mean of the liquid-equivalent precipitation particle diameter (mm).

count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
  -9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
  -9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
  -9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
  -9999 Missing value

precipTotLogNw (Group in G1)
Common logarithm of the intercept of the normalized gamma distribution representing the liquid-equivalent precipitation particle size distribution (log10(m⁻⁴)).
count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

precipAllObs (4-byte integer, array size: ltL x lnL x ns x hgt x st):
Number of total observations, whether precipitating or not. Special values are defined as:
-9999 Missing value

surfPrecipTotRateDiurnal (Group in G1)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitating water in the
lowest uncontaminated range-bin (mm/hr), indexed by the local time. (Note: liquid can
be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of
ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x ns x st x tim):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x st x tim):
Mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x st x tim):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9 Missing value

surfPrecipTotRateDiurnalAllObs (4-byte integer, array size: ltL x lnL x ns x st x tim):
Number of total diurnal observations, whether precipitating or not. Special values are
defined as:
-9999 Missing value
surfPrecipTotRateUn (4-byte float, array size: ltL x lnL x ns):
Surface total precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9 Missing value

surfPrecipLiqRateUn (4-byte float, array size: ltL x lnL x ns):
Surface liquid precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9 Missing value

surfPrecipTotRateProb (4-byte float, array size: ltL x lnL x ns):
Probability of total surface precipitation. Special values are defined as:
-9999.9 Missing value

surfPrecipLiqRateProb (4-byte float, array size: ltL x lnL x ns):
Probability of liquid surface precipitation. Special values are defined as:
-9999.9 Missing value

G2 (Grid)

G2_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipTotRate (Group in G2)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitation water (mm/hr).
(Nota: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value
**precipLiqRate** (Group in G2)
Equivalent precipitation rate of liquid-phase precipitating water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipTotWaterContent** (Group in G2)
Equivalent water content of both liquid-phase and ice-phase precipitating water ($g/m^3$). (Note: liquid can be in the form of rain or melt water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipLiqWaterContent** (Group in G2)
Equivalent water content of liquid-phase precipitating water ($g/m^3$). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value
**stdev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**cloudLiqWaterContent** (Group in G2)
Equivalent water content of liquid-phase cloud water ($g/m^3$).

**count** (4-byte integer, array size: ltH x lnH x ns x hgt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt):
Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x hgt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipTotDm** (Group in G2)
Volume-weighted mean of the liquid-equivalent precipitation particle diameter (mm).

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipTotLogNw** (Group in G2)
Common logarithm of the intercept of the normalized gamma distribution representing the liquid-equivalent precipitation particle size distribution ($\log_{10}(m^{-4})$).

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value
mean (4-byte float, array size: $ltH \times lnH \times ns \times hgt \times rt$):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: $ltH \times lnH \times ns \times hgt \times rt$):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9  Missing value

precipAllObs (4-byte integer, array size: $ltH \times lnH \times ns \times hgt$):
Number of total observations, whether precipitating or not. Special values are defined as:
-9999  Missing value

surfPrecipTotRateDiurnal (Group in G2)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitating water in the
lowest uncontaminated range-bin (mm/hr), indexed by the local time. (Note: liquid can
be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of
ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: $ltH \times lnH \times ns \times tim$):
Count. Special values are defined as:
-9999  Missing value

mean (4-byte float, array size: $ltH \times lnH \times ns \times tim$):
mean. Special values are defined as:
-9999.9  Missing value

stdev (4-byte float, array size: $ltH \times lnH \times ns \times tim$):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
-9999.9  Missing value

surfPrecipTotRateDiurnalAllObs (4-byte integer, array size: $ltH \times lnH \times ns \times tim$):
Number of total diurnal observations, whether precipitating or not. Special values are
defined as:
-9999  Missing value

surfPrecipTotRateUn (4-byte float, array size: $ltH \times lnH \times ns$):
Surface total precipitation rate unconditioned. To obtain rate conditioned on precipita-
tion, divide by the probability. Special values are defined as:
-9999.9  Missing value

surfPrecipLiqRateUn (4-byte float, array size: $ltH \times lnH \times ns$):
Surface liquid precipitation rate unconditioned. To obtain rate conditioned on precipita-
tion, divide by the probability. Special values are defined as:
-9999.9  Missing value
surfPrecipTotRateProb (4-byte float, array size: lH x lnH x ns):
Probability of total surface precipitation. Special values are defined as:
-9999.9 Missing value

surfPrecipLiqRateProb (4-byte float, array size: lH x lnH x ns):
Probability of liquid surface precipitation. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_3CMBX_H_
define _TK_3CMBX_H_

#ifndef _L3CMBX_G2_SURFPRECIPTOTRATEDIURNAL_
define _L3CMBX_G2_SURFPRECIPTOTRATEDIURNAL_

typedef struct {
    int count[24][4][1440][536];
    float mean[24][4][1440][536];
    float stdev[24][4][1440][536];
} L3CMBX_G2_SURFPRECIPTOTRATEDIURNAL;
#endif

#ifndef _L3CMBX_G2_PRECIPTOTLOGNW_
define _L3CMBX_G2_PRECIPTOTLOGNW_

typedef struct {
    int count[3][16][4][1440][536];
    float mean[3][16][4][1440][536];
    float stdev[3][16][4][1440][536];
} L3CMBX_G2_PRECIPTOTLOGNW;
#endif

#ifndef _L3CMBX_G2_PRECIPTOTDM_
define _L3CMBX_G2_PRECIPTOTDM_

typedef struct {
    int count[3][16][4][1440][536];
    float mean[3][16][4][1440][536];
    float stdev[3][16][4][1440][536];
} L3CMBX_G2_PRECIPTOTDM;
```
typedef struct {
    int count[16][4][1440][536];
    float mean[16][4][1440][536];
    float stdev[16][4][1440][536];
} L3CMBX_G2_CLOUDLIQWATERCONTENT;

typedef struct {
    int count[3][16][4][1440][536];
    float mean[3][16][4][1440][536];
    float stdev[3][16][4][1440][536];
} L3CMBX_G2_PRECIPLIQWATERCONTENT;

typedef struct {
    int count[3][16][4][1440][536];
    float mean[3][16][4][1440][536];
    float stdev[3][16][4][1440][536];
} L3CMBX_G2_PRECIPTOTWATERCONTENT;

typedef struct {
    int count[3][16][4][1440][536];
    float mean[3][16][4][1440][536];
    float stdev[3][16][4][1440][536];
} L3CMBX_G2_PRECIPLIQRATE;
# ifndef _L3CMBX_G2_PRECIPTOTRATE_
#define _L3CMBX_G2_PRECIPTOTRATE_

typedef struct {
    int count[3][16][4][1440][536];
    float mean[3][16][4][1440][536];
    float stdev[3][16][4][1440][536];
} L3CMBX_G2_PRECIPTOTRATE;

#endif

#endif

#ifndef _L3CMBX_G2_
#define _L3CMBX_G2_

typedef struct {
    L3CMBX_G2_PRECIPTOTRATE precipTotRate;
    L3CMBX_G2_PRECIPLIQRATE precipLiqRate;
    L3CMBX_G2_PRECIPTOTWATERCONTENT precipTotWaterContent;
    L3CMBX_G2_PRECIPLIQWATERCONTENT precipLiqWaterContent;
    L3CMBX_G2_CLOUDLIQWATERCONTENT cloudLiqWaterContent;
    L3CMBX_G2_PRECIPTOTDM precipTotDm;
    L3CMBX_G2_PRECIPTOTLOGNW precipTotLogNw;
    int precipAllObs[16][4][1440][536];
    int surfPrecipTotRateDiurnalAllObs[24][4][1440][536];
    float surfPrecipTotRateUn[4][1440][536];
    float surfPrecipLiqRateUn[4][1440][536];
    float surfPrecipTotRateProb[4][1440][536];
    float surfPrecipLiqRateProb[4][1440][536];
} L3CMBX_G2;

#endif

#ifndef _L3CMBX_G1_SURFPRECIPTOTRATEDIURNAL_
#define _L3CMBX_G1_SURFPRECIPTOTRATEDIURNAL_

typedef struct {
    int count[24][3][4][72][28];
    float mean[24][3][4][72][28];
    float stdev[24][3][4][72][28];
} L3CMBX_G1_SURFPRECIPTOTRATEDIURNAL;

#endif
} L3CMBX_G1_SURFPRECIPTOTRATEDIURNAL;
#endif

#ifndef _L3CMBX_G1_PRECIPTOTLOGNW_
#define _L3CMBX_G1_PRECIPTOTLOGNW_

typedef struct {
    int count[3][3][16][4][72][28];
    float mean[3][3][16][4][72][28];
    float stdev[3][3][16][4][72][28];
    int hist[30][3][3][16][4][72][28];
} L3CMBX_G1_PRECIPTOTLOGNW;
#endif

#ifndef _L3CMBX_G1_PRECIPTOTDM_
#define _L3CMBX_G1_PRECIPTOTDM_

typedef struct {
    int count[3][3][16][4][72][28];
    float mean[3][3][16][4][72][28];
    float stdev[3][3][16][4][72][28];
    int hist[30][3][3][16][4][72][28];
} L3CMBX_G1_PRECIPTOTDM;
#endif

#ifndef _L3CMBX_G1_CLOUDLIQWATERCONTENT_
#define _L3CMBX_G1_CLOUDLIQWATERCONTENT_

typedef struct {
    int count[3][16][4][72][28];
    float mean[3][16][4][72][28];
    float stdev[3][16][4][72][28];
    int hist[30][3][16][4][72][28];
} L3CMBX_G1_CLOUDLIQWATERCONTENT;
#endif

#ifndef _L3CMBX_G1_PRECIPLIQWATERCONTENT_
#define _L3CMBX_G1_PRECIPLIQWATERCONTENT_

typedef struct {
    int count[3][16][4][72][28];
    float mean[3][16][4][72][28];
    float stdev[3][16][4][72][28];
    int hist[30][3][16][4][72][28];
} L3CMBX_G1_PRECIPLIQWATERCONTENT;
#endif
typedef struct {
    int count[3][3][16][4][72][28];
    float mean[3][3][16][4][72][28];
    float stdev[3][3][16][4][72][28];
    int hist[30][3][3][16][4][72][28];
} L3CMBX_G1_PRECIIQWATERCONTENT;
#endif

#ifndef _L3CMBX_G1_PRECIIQWATERCONTENT_
#define _L3CMBX_G1_PRECIIQWATERCONTENT_

typedef struct {
    int count[3][3][16][4][72][28];
    float mean[3][3][16][4][72][28];
    float stdev[3][3][16][4][72][28];
    int hist[30][3][3][16][4][72][28];
} L3CMBX_G1_PRECIIQWATERCONTENT;
#endif

#ifndef _L3CMBX_G1_PRECIIQRATE_
#define _L3CMBX_G1_PRECIIQRATE_

typedef struct {
    int count[3][3][16][4][72][28];
    float mean[3][3][16][4][72][28];
    float stdev[3][3][16][4][72][28];
    int hist[30][3][3][16][4][72][28];
} L3CMBX_G1_PRECIIQRATE;
#endif

#ifndef _L3CMBX_G1_PRECIIQTOTRATE_
#define _L3CMBX_G1_PRECIIQTOTRATE_

typedef struct {
    int count[3][3][16][4][72][28];
    float mean[3][3][16][4][72][28];
    float stdev[3][3][16][4][72][28];
    int hist[30][3][3][16][4][72][28];
} L3CMBX_G1_PRECIIQTOTRATE;
```c
#define _L3CMBX_G1_

typedef struct {
    L3CMBX_G1_PRECIPTOTRATE precipTotRate;
    L3CMBX_G1_PRECIPLIQRATE precipLiqRate;
    L3CMBX_G1_PRECIPTOTWATERCONTENT precipTotWaterContent;
    L3CMBX_G1_PRECIPLIQWATERCONTENT precipLiqWaterContent;
    L3CMBX_G1_CLOUDLIQWATERCONTENT cloudLiqWaterContent;
    L3CMBX_G1_PRECIPTOTDM precipTotDm;
    L3CMBX_G1_PRECIPTOTLOGNW precipTotLogNw;
    int precipAllObs[3][16][4][72][28];
    L3CMBX_G1_SURFPRECIPTOTRATEDIURNAL surfPrecipTotRateDiurnal;
    int surfPrecipTotRateDiurnalAllObs[24][3][4][72][28];
    float surfPrecipTotRateUn[4][72][28];
    float surfPrecipLiqRateUn[4][72][28];
    float surfPrecipTotRateProb[4][72][28];
    float surfPrecipLiqRateProb[4][72][28];
} L3CMBX_G1;

#define _L3CMBX_GRIDS_

typedef struct {
    L3CMBX_G1 G1;
    L3CMBX_G2 G2;
} L3CMBX_GRIDS;
```

**Fortran Structure Header file:**

```fortran
STRUCTURE /L3CMBX_G2_SURFPRECIPTOTRATEDIURNAL/
    INTEGER*4 count(536,1440,4,24)
    REAL*4 mean(536,1440,4,24)
    REAL*4 stdev(536,1440,4,24)
END STRUCTURE
```
STRUCTURE /L3CMBX_G2_PRECIPTOTLOGNW/
   INTEGER*4 count(536,1440,4,16,3)
   REAL*4 mean(536,1440,4,16,3)
   REAL*4 stdev(536,1440,4,16,3)
END STRUCTURE

STRUCTURE /L3CMBX_G2_PRECIPTOTDM/
   INTEGER*4 count(536,1440,4,16,3)
   REAL*4 mean(536,1440,4,16,3)
   REAL*4 stdev(536,1440,4,16,3)
END STRUCTURE

STRUCTURE /L3CMBX_G2_CLOUDLIQWATERCONTENT/
   INTEGER*4 count(536,1440,4,16)
   REAL*4 mean(536,1440,4,16)
   REAL*4 stdev(536,1440,4,16)
END STRUCTURE

STRUCTURE /L3CMBX_G2_PRECIPLIQWATERCONTENT/
   INTEGER*4 count(536,1440,4,16,3)
   REAL*4 mean(536,1440,4,16,3)
   REAL*4 stdev(536,1440,4,16,3)
END STRUCTURE

STRUCTURE /L3CMBX_G2_PRECIPTOTWATERCONTENT/
   INTEGER*4 count(536,1440,4,16,3)
   REAL*4 mean(536,1440,4,16,3)
   REAL*4 stdev(536,1440,4,16,3)
END STRUCTURE

STRUCTURE /L3CMBX_G2_PRECIPLIQRATE/
   INTEGER*4 count(536,1440,4,16,3)
   REAL*4 mean(536,1440,4,16,3)
   REAL*4 stdev(536,1440,4,16,3)
END STRUCTURE

STRUCTURE /L3CMBX_G2_PRECIPTOTRATE/
   INTEGER*4 count(536,1440,4,16,3)
   REAL*4 mean(536,1440,4,16,3)
   REAL*4 stdev(536,1440,4,16,3)
END STRUCTURE
STRUCTURE /L3CMBX_G2/
  RECORD /L3CMBX_G2_PRECIPTOTRATE/ precipTotRate
  RECORD /L3CMBX_G2_PRECIPLIQRATE/ precipLiqRate
  RECORD /L3CMBX_G2_PRECIPTOTWATERCONTENT/ precipTotWaterContent
  RECORD /L3CMBX_G2_PRECIPLIQWATERCONTENT/ precipLiqWaterContent
  RECORD /L3CMBX_G2_CLOUDLIQWATERCONTENT/ cloudLiqWaterContent
  RECORD /L3CMBX_G2_PRECIPTOTDM/ precipTotDm
  RECORD /L3CMBX_G2_PRECIPTOTLOGNW/ precipTotLogNw
  INTEGER*4 precipAllObs(536,1440,4,16)
  RECORD /L3CMBX_G2_SURFPRECIPTOTRATEDIURNAL/ surfPrecipTotRateDiurnal
  INTEGER*4 surfPrecipTotRateDiurnalAllObs(536,1440,4,24)
  REAL*4 surfPrecipTotRateUn(536,1440,4)
  REAL*4 surfPrecipLiqRateUn(536,1440,4)
  REAL*4 surfPrecipTotRateProb(536,1440,4)
  REAL*4 surfPrecipLiqRateProb(536,1440,4)
END STRUCTURE

STRUCTURE /L3CMBX_G1_SURFPRECIPTOTRATEDIURNAL/
  INTEGER*4 count(28,72,4,3,24)
  REAL*4 mean(28,72,4,3,24)
  REAL*4 stdev(28,72,4,3,24)
END STRUCTURE

STRUCTURE /L3CMBX_G1_PRECIPTOTLOGNW/
  INTEGER*4 count(28,72,4,16,3,3)
  REAL*4 mean(28,72,4,16,3,3)
  REAL*4 stdev(28,72,4,16,3,3)
  INTEGER*4 hist(28,72,4,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBX_G1_PRECIPTOTDM/
  INTEGER*4 count(28,72,4,16,3,3)
  REAL*4 mean(28,72,4,16,3,3)
  REAL*4 stdev(28,72,4,16,3,3)
  INTEGER*4 hist(28,72,4,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBX_G1_CLOUDLIQWATERCONTENT/
  INTEGER*4 count(28,72,4,16,3)
  REAL*4 mean(28,72,4,16,3)
  REAL*4 stdev(28,72,4,16,3)
  INTEGER*4 hist(28,72,4,16,3,30)
END STRUCTURE
STRUCTURE /L3CMBX_G1_PRECIPLIQWATERCONTENT/
    INTEGER*4 count(28,72,4,16,3,3)
    REAL*4 mean(28,72,4,16,3,3)
    REAL*4 stdev(28,72,4,16,3,3)
    INTEGER*4 hist(28,72,4,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBX_G1_PRECIPTOTWATERCONTENT/
    INTEGER*4 count(28,72,4,16,3,3)
    REAL*4 mean(28,72,4,16,3,3)
    REAL*4 stdev(28,72,4,16,3,3)
    INTEGER*4 hist(28,72,4,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBX_G1_PRECIPLIQWATERCONTENT/
    INTEGER*4 count(28,72,4,16,3,3)
    REAL*4 mean(28,72,4,16,3,3)
    REAL*4 stdev(28,72,4,16,3,3)
    INTEGER*4 hist(28,72,4,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBX_G1_PRECIPTOTRATE/
    INTEGER*4 count(28,72,4,16,3,3)
    REAL*4 mean(28,72,4,16,3,3)
    REAL*4 stdev(28,72,4,16,3,3)
    INTEGER*4 hist(28,72,4,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBX_G1/
    RECORD /L3CMBX_G1_PRECIPTOTRATE/ precipTotRate
    RECORD /L3CMBX_G1_PRECIPLIQWATERCONTENT/ precipLiqRate
    RECORD /L3CMBX_G1_PRECIPTOTWATERCONTENT/ precipTotWaterContent
    RECORD /L3CMBX_G1_CLOUDLIQWATERCONTENT/ cloudLiqWaterContent
    RECORD /L3CMBX_G1_PRECIPTOTDM/ precipTotDm
    RECORD /L3CMBX_G1_PRECIPTOTLOGNW/ precipTotLogNw
    INTEGER*4 precipAllObs(28,72,4,16,3)
    RECORD /L3CMBX_G1_SURFPRECIPTOTRATEDIURNAL/ surfPrecipTotRateDiurnal
    INTEGER*4 surfPrecipTotRateDiurnalAllObs(28,72,4,3,24)
    REAL*4 surfPrecipTotRateUn(28,72,4)
    REAL*4 surfPrecipLiqRateUn(28,72,4)
    REAL*4 surfPrecipTotRateProb(28,72,4)
REAL*4 surfPrecipLiqRateProb(28,72,4)
END STRUCTURE

STRUCTURE /L3CMBX_GRIDS/
  RECORD /L3CMBX_G1/ G1
  RECORD /L3CMBX_G2/ G2
END STRUCTURE

5.72 3CMBTX - Combined precipitation

3CMBT, “Combined precipitation”, computes statistics of the Combined measurements at both a low horizontal resolution (G1, 5° x 5° latitude/longitude) and a high horizontal resolution (G2, 0.25° x 0.25° latitude/longitude). There will be both a monthly product and a daily product.

Units and ranges not included in this version. When units and ranges are provided and no more changes are coming then they could be added. Use specific reference for each variable.

Dimension definitions:

ltL  28  Number of low resolution 5° grid intervals of latitude from 70°S to 70°N.
lnL  72  Number of low resolution 5° grid intervals of longitude from 180°W to 180°E.
ltH  536 Number of high resolution 0.25° grid intervals of latitude from 67°S to 67°N.
lnH 1440 Number of high resolution 0.25° grid intervals of longitude from 180°W to 180°E.
ns  2   Number of swaths: 0 = Ku+microwave (FS), 1 = Ku+microwave narrow swath (MS).
ht  16   Number of level heights 0-15: 0: near surface, 1-10: height = 1.0km * index, 11-15: height = 10.0km + 2.0km * (index-10),
tim  24   Number of hourly local time bins.
rt  3   Number of rain types: stratiform, convective, all.
st  3   Number of surface types: ocean, land, all.
bin 30  Number of bins in histogram.

Figure 1109 through Figure 1127 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.
5.72 3CMBTX - Combined precipitation

Figure 1109: Data Format Structure for 3CMBTX, Combined precipitation

Figure 1110: Data Format Structure for 3CMBTX, G1
Figure 1111: Data Format Structure for 3CMBTX, G2

Figure 1112: Data Format Structure for 3CMBTX, G1, precipTotRate

Figure 1113: Data Format Structure for 3CMBTX, G1, precipLiqRate
5.72 3CMBTX - Combined precipitation

Figure 1114: Data Format Structure for 3CMBTX, G1, precipTotWaterContent

Figure 1115: Data Format Structure for 3CMBTX, G1, precipLiqWaterContent

Figure 1116: Data Format Structure for 3CMBTX, G1, cloudLiqWaterContent

Figure 1117: Data Format Structure for 3CMBTX, G1, precipTotDm

Figure 1118: Data Format Structure for 3CMBTX, G1, precipTotLogNw

Figure 1119: Data Format Structure for 3CMBTX, G1, surfPrecipTotRateDiurnal
Figure 1120: Data Format Structure for 3CMBTX, G2, precipTotRate
5.72 3CMBTX - Combined precipitation

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputFileNames** (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

**InputAlgorithmVersions** (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

**InputGenerationDateTimes** (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**Grids** (Group)
Figure 1124: Data Format Structure for 3CMBTX, G2, cloudLiqWaterContent

Figure 1125: Data Format Structure for 3CMBTX, G2, precipTotDm

Figure 1126: Data Format Structure for 3CMBTX, G2, precipTotLogNw

Figure 1127: Data Format Structure for 3CMBTX, G2, surfPrecipTotRateDiurnal
G1 (Grid)

**G1_GridHeader** (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**precipTotRate** (Group in G1)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitation water (mm/hr).
(Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipLiqRate** (Group in G1)
Equivalent precipitation rate of liquid-phase precipitating water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

**count** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value
hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
  -9999 Missing value

precipTotWaterContent (Group in G1)
Equivalent water content of both liquid-phase and ice-phase precipitating water (g/m^3).
(Note: liquid can be in the form of rain or melt water in mixed-phase particles; ice can
be in the form of ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
  -9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
  -9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
  -9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
  -9999 Missing value

precipLiqWaterContent (Group in G1)
Equivalent water content of liquid-phase precipitating water (g/m^3). (Note: liquid can
be in the form of rain or liquid water in mixed-phase particles.)

count (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
  -9999 Missing value

mean (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
  -9999.9 Missing value

stdev (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
  -9999.9 Missing value

hist (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
  -9999 Missing value
**cloudLiqWaterContent** (Group in G1)
Equivalent water content of liquid-phase cloud water (g/m³).

**count** (4-byte integer, array size: ltL x lnL x ns x hgt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x hgt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x hgt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x ns x hgt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipTotDm** (Group in G1)
Volume-weighted mean of the liquid-equivalent precipitation particle diameter (mm).

**count** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipTotLogNw** (Group in G1)
Common logarithm of the intercept of the normalized gamma distribution representing the liquid-equivalent precipitation particle size distribution (log₁₀(m⁻⁴)).
**count** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x hgt x rt x st):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**hist** (4-byte integer, array size: ltL x lnL x ns x hgt x rt x st x bin):
Histogram. Special values are defined as:
-9999 Missing value

**precipAllObs** (4-byte integer, array size: ltL x lnL x ns x hgt x st):
Number of total observations, whether precipitating or not. Special values are defined as:
-9999 Missing value

**surfPrecipTotRateDiurnal** (Group in G1)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitating water in the lowest uncontaminated range-bin (mm/hr), indexed by the local time. (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltL x lnL x ns x st x tim):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltL x lnL x ns x st x tim):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltL x lnL x ns x st x tim):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**surfPrecipTotRateDiurnalAllObs** (4-byte integer, array size: ltL x lnL x ns x st x tim):
Number of total diurnal observations, whether precipitating or not. Special values are defined as:
-9999 Missing value
surfPrecipTotRateUn (4-byte float, array size: ltL x lnL x ns):
Surface total precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9 Missing value

surfPrecipLiqRateUn (4-byte float, array size: ltL x lnL x ns):
Surface liquid precipitation rate unconditioned. To obtain rate conditioned on precipitation, divide by the probability. Special values are defined as:
-9999.9 Missing value

surfPrecipTotRateProb (4-byte float, array size: ltL x lnL x ns):
Probability of total surface precipitation. Special values are defined as:
-9999.9 Missing value

surfPrecipLiqRateProb (4-byte float, array size: ltL x lnL x ns):
Probability of liquid surface precipitation. Special values are defined as:
-9999.9 Missing value

G2 (Grid)

G2_GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

precipTotRate (Group in G2)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitation water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)
count (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

mean (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

stdev (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value
**precipLiqRate** (Group in G2)
Equivalent precipitation rate of liquid-phase precipitating water (mm/hr). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipTotWaterContent** (Group in G2)
Equivalent water content of both liquid-phase and ice-phase precipitating water ($g/m^3$). (Note: liquid can be in the form of rain or melt water in mixed-phase particles; ice can be in the form of ice particles or ice in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stdev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipLiqWaterContent** (Group in G2)
Equivalent water content of liquid-phase precipitating water ($g/m^3$). (Note: liquid can be in the form of rain or liquid water in mixed-phase particles.)

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value
**stddev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**cloudLiqWaterContent** (Group in G2)
Equivalent water content of liquid-phase cloud water ($g/m^3$).

**count** (4-byte integer, array size: ltH x lnH x ns x hgt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt):
mean. Special values are defined as:
-9999.9 Missing value

**stddev** (4-byte float, array size: ltH x lnH x ns x hgt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipTotDm** (Group in G2)
Volume-weighted mean of the liquid-equivalent precipitation particle diameter (mm).

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value

**mean** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
mean. Special values are defined as:
-9999.9 Missing value

**stddev** (4-byte float, array size: ltH x lnH x ns x hgt x rt):
Standard deviation for the monthly product. Mean of squares for the daily product. Special values are defined as:
-9999.9 Missing value

**precipTotLogNw** (Group in G2)
Common logarithm of the intercept of the normalized gamma distribution representing the liquid-equivalent precipitation particle size distribution ($\log_{10}(m^{-4})$).

**count** (4-byte integer, array size: ltH x lnH x ns x hgt x rt):
Count. Special values are defined as:
-9999 Missing value
mean (4-byte float, array size: \(ltH \times lnH \times ns \times hgt \times rt\)):
mean. Special values are defined as:
  \(-9999.9\) Missing value

stdev (4-byte float, array size: \(ltH \times lnH \times ns \times hgt \times rt\)):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
  \(-9999.9\) Missing value

precipAllObs (4-byte integer, array size: \(ltH \times lnH \times ns \times hgt\)):
Number of total observations, whether precipitating or not. Special values are defined as:
  \(-9999\) Missing value

surfPrecipTotRateDiurnal (Group in G2)
Equivalent precipitation rate of both liquid-phase and ice-phase precipitating water in the
lowest uncontaminated range-bin (mm/hr), indexed by the local time. (Note: liquid can
be in the form of rain or liquid water in mixed-phase particles; ice can be in the form of
ice particles or ice in mixed-phase particles.)

count (4-byte integer, array size: \(ltH \times lnH \times ns \times tim\)):
Count. Special values are defined as:
  \(-9999\) Missing value

mean (4-byte float, array size: \(ltH \times lnH \times ns \times tim\)):
mean. Special values are defined as:
  \(-9999.9\) Missing value

stdev (4-byte float, array size: \(ltH \times lnH \times ns \times tim\)):
Standard deviation for the monthly product. Mean of squares for the daily product.
Special values are defined as:
  \(-9999.9\) Missing value

surfPrecipTotRateDiurnalAllObs (4-byte integer, array size: \(ltH \times lnH \times ns \times tim\)):
Number of total diurnal observations, whether precipitating or not. Special values are
defined as:
  \(-9999\) Missing value

surfPrecipTotRateUn (4-byte float, array size: \(ltH \times lnH \times ns\)):
Surface total precipitation rate unconditioned. To obtain rate conditioned on precipita-
tion, divide by the probability. Special values are defined as:
  \(-9999.9\) Missing value

surfPrecipLiqRateUn (4-byte float, array size: \(ltH \times lnH \times ns\)):
Surface liquid precipitation rate unconditioned. To obtain rate conditioned on precipita-
tion, divide by the probability. Special values are defined as:
  \(-9999.9\) Missing value
surfPrecipTotRateProb (4-byte float, array size: ltH x lnH x ns):
Probability of total surface precipitation. Special values are defined as:
-9999.9  Missing value

surfPrecipLiqRateProb (4-byte float, array size: ltH x lnH x ns):
Probability of liquid surface precipitation. Special values are defined as:
-9999.9  Missing value

C Structure Header file:

```c
#ifndef _TK_3CMBTX_H_
define _TK_3CMBTX_H_

#ifndef _L3CMBTX_G2_SURFPRECIPTOTRATEDIURNAL_
define _L3CMBTX_G2_SURFPRECIPTOTRATEDIURNAL_

typedef struct {
    int count[24][2][1440][536];
    float mean[24][2][1440][536];
    float stdev[24][2][1440][536];
} L3CMBTX_G2_SURFPRECIPTOTRATEDIURNAL;
#endif

#ifndef _L3CMBTX_G2_PRECIPTOTLOGNW_
define _L3CMBTX_G2_PRECIPTOTLOGNW_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMBTX_G2_PRECIPTOTLOGNW;
#endif

#ifndef _L3CMBTX_G2_PRECIPTOTDM_
define _L3CMBTX_G2_PRECIPTOTDM_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMBTX_G2_PRECIPTOTDM;
```
#ifndef _L3CMBTX_G2_CLOUDLIQWATERCONTENT_
#define _L3CMBTX_G2_CLOUDLIQWATERCONTENT_

typedef struct {
    int count[16][2][1440][536];
    float mean[16][2][1440][536];
    float stdev[16][2][1440][536];
} L3CMBTX_G2_CLOUDLIQWATERCONTENT;
#endif

#ifndef _L3CMBTX_G2_PRECIPLIQWATERCONTENT_
#define _L3CMBTX_G2_PRECIPLIQWATERCONTENT_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMBTX_G2_PRECIPLIQWATERCONTENT;
#endif

#ifndef _L3CMBTX_G2_PRECIPTOTWATERCONTENT_
#define _L3CMBTX_G2_PRECIPTOTWATERCONTENT_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMBTX_G2_PRECIPTOTWATERCONTENT;
#endif

#ifndef _L3CMBTX_G2_PRECIPLIQRATE_
#define _L3CMBTX_G2_PRECIPLIQRATE_

typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMBTX_G2_PRECIPLIQRATE;
typedef struct {
    int count[3][16][2][1440][536];
    float mean[3][16][2][1440][536];
    float stdev[3][16][2][1440][536];
} L3CMBTX_G2_PRECIPTOTRATE;

typedef struct {
    L3CMBTX_G2_PRECIPTOTRATE precipTotRate;
    L3CMBTX_G2_PRECIPLIQRATE precipLiqRate;
    L3CMBTX_G2_PRECIPTOTWATERCONTENT precipTotWaterContent;
    L3CMBTX_G2_PRECIPLIQWATERCONTENT precipLiqWaterContent;
    L3CMBTX_G2_CLOUDLIQWATERCONTENT cloudLiqWaterContent;
    L3CMBTX_G2_PRECIPTOTDM precipTotDm;
    L3CMBTX_G2_PRECIPTOTLOGNW precipTotLogNw;
    int precipAllObs[16][2][1440][536];
    int surfPrecipTotRateDiurnalAllObs[24][2][1440][536];
    float surfPrecipTotRateUn[2][1440][536];
    float surfPrecipLiqRateUn[2][1440][536];
    float surfPrecipTotRateProb[2][1440][536];
    float surfPrecipLiqRateProb[2][1440][536];
} L3CMBTX_G2;

typedef struct {
    int count[24][3][2][72][28];
    float mean[24][3][2][72][28];
    float stdev[24][3][2][72][28];
} L3CMBTX_G1_SURFPRECIPTOTRATEDIURNAL;
} L3CMBTX_G1_SURFPRECIPTOTRATEDIURNAL;

#endif

#ifndef _L3CMBTX_G1_PRECIPTOTLOGNW_
#define _L3CMBTX_G1_PRECIPTOTLOGNW_

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMBTX_G1_PRECIPTOTLOGNW;

#endif

#ifndef _L3CMBTX_G1_PRECIPTOTDM_
#define _L3CMBTX_G1_PRECIPTOTDM_

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMBTX_G1_PRECIPTOTDM;

#endif

#ifndef _L3CMBTX_G1_CLOUDLIQWATERCONTENT_
#define _L3CMBTX_G1_CLOUDLIQWATERCONTENT_

typedef struct {
    int count[3][16][2][72][28];
    float mean[3][16][2][72][28];
    float stdev[3][16][2][72][28];
    int hist[30][3][16][2][72][28];
} L3CMBTX_G1_CLOUDLIQWATERCONTENT;

#endif

#ifndef _L3CMBTX_G1_PRECIPLIQWATERCONTENT_
#define _L3CMBTX_G1_PRECIPLIQWATERCONTENT_

typedef struct {
    int count[3][16][2][72][28];
    float mean[3][16][2][72][28];
    float stdev[3][16][2][72][28];
    int hist[30][3][16][2][72][28];
} L3CMBTX_G1_PRECIPLIQWATERCONTENT;

#endif
typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMBTX_G1_PRECIPLIQRATE;

#endif

#ifndef _L3CMBTX_G1_PRECIPTOTRATE_
#define _L3CMBTX_G1_PRECIPTOTRATE_

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMBTX_G1_PRECIPTOTRATE;

#endif

#ifndef _L3CMBTX_G1_PRECIPLIQRATE_ 
#define _L3CMBTX_G1_PRECIPLIQRATE_ 

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMBTX_G1_PRECIPLIQRATE;

#endif

#ifndef _L3CMBTX_G1_PRECIPTOTWATERCONTENT_
#define _L3CMBTX_G1_PRECIPTOTWATERCONTENT_

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMBTX_G1_PRECIPTOTWATERCONTENT;

#endif

#ifndef _L3CMBTX_G1_PRECIPTOTWATERCONTENT_
#define _L3CMBTX_G1_PRECIPTOTWATERCONTENT_

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMBTX_G1_PRECIPTOTWATERCONTENT;

#endif

#ifndef _L3CMBTX_G1_PRECIPTOTWATERCONTENT_
#define _L3CMBTX_G1_PRECIPTOTWATERCONTENT_

typedef struct {
    int count[3][3][16][2][72][28];
    float mean[3][3][16][2][72][28];
    float stdev[3][3][16][2][72][28];
    int hist[30][3][3][16][2][72][28];
} L3CMBTX_G1_PRECIPTOTWATERCONTENT;
typedef struct {
    L3CMBTX_G1_PRECIPTOTRATE precipTotRate;
    L3CMBTX_G1_PRECIPLIQRATE precipLiqRate;
    L3CMBTX_G1_PRECIPTOTWATERCONTENT precipTotWaterContent;
    L3CMBTX_G1_PRECIPLIQWATERCONTENT precipLiqWaterContent;
    L3CMBTX_G1_CLOUDLIQWATERCONTENT cloudLiqWaterContent;
    L3CMBTX_G1_PRECIPTOTDM precipTotDm;
    L3CMBTX_G1_PRECIPTOTLOGNW precipTotLogNw;
    int precipAllObs[3][16][2][72][28];
    L3CMBTX_G1_SURFPRECIPTOTRATEDIURNAL surfPrecipTotRateDiurnal;
    int surfPrecipTotRateDiurnalAllObs[24][3][2][72][28];
    float surfPrecipTotRateUn[2][72][28];
    float surfPrecipLiqRateUn[2][72][28];
    float surfPrecipTotRateProb[2][72][28];
    float surfPrecipLiqRateProb[2][72][28];
} L3CMBTX_G1;

typedef struct {
    L3CMBTX_G1 G1;
    L3CMBTX_G2 G2;
} L3CMBTX_GRIDS;

Fortran Structure Header file:

STRUCTURE /L3CMBTX_G2_SURFPRECIPTOTRATEDIURNAL/
    INTEGER*4 count(536,1440,2,24)
    REAL*4 mean(536,1440,2,24)
    REAL*4 stdev(536,1440,2,24)
END STRUCTURE
STRUCTURE /L3CMBTX_G2_PRECIPTOTLOGNW/
  INTEGER*4 count(536,1440,2,16,3)
  REAL*4 mean(536,1440,2,16,3)
  REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMBTX_G2_PRECIPTOTDM/
  INTEGER*4 count(536,1440,2,16,3)
  REAL*4 mean(536,1440,2,16,3)
  REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMBTX_G2_CLOUDLIQWATERCONTENT/
  INTEGER*4 count(536,1440,2,16)
  REAL*4 mean(536,1440,2,16)
  REAL*4 stdev(536,1440,2,16)
END STRUCTURE

STRUCTURE /L3CMBTX_G2_PRECIIQWATERCONTENT/
  INTEGER*4 count(536,1440,2,16,3)
  REAL*4 mean(536,1440,2,16,3)
  REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMBTX_G2_PRECIPTOTWATERCONTENT/
  INTEGER*4 count(536,1440,2,16,3)
  REAL*4 mean(536,1440,2,16,3)
  REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMBTX_G2_PRECIPLIQRATE/
  INTEGER*4 count(536,1440,2,16,3)
  REAL*4 mean(536,1440,2,16,3)
  REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE

STRUCTURE /L3CMBTX_G2_PRECIPTOTRATE/
  INTEGER*4 count(536,1440,2,16,3)
  REAL*4 mean(536,1440,2,16,3)
  REAL*4 stdev(536,1440,2,16,3)
END STRUCTURE
STRUCTURE /L3CMBTX_G2/
  RECORD /L3CMBTX_G2_PRECIPTOTRATE/ precipTotRate
  RECORD /L3CMBTX_G2_PRECIPL IQRATE/ precipLiqRate
  RECORD /L3CMBTX_G2_PRECIPTOTW ATERCONTENT/ precipTotWaterContent
  RECORD /L3CMBTX_G2_PRECIPL IQWATERCONTENT/ precipLiqWaterContent
  RECORD /L3CMBTX_G2_CLOUDLIQWATERCONTENT/ cloudLiqWaterContent
  RECORD /L3CMBTX_G2_PRECIPTOTDM/ precipTotDm
  RECORD /L3CMBTX_G2_PRECIPTOTLOGNW/ precipTotLogNw
  INTEGER*4 precipAllObs(536,1440,2,16)
  RECORD /L3CMBTX_G2_SURFPRECIPTOTRATEDIURNAL/ surfPrecipTotRateDiurnal
    INTEGER*4 surfPrecipTotRateDiurnalAllObs(536,1440,2,24)
    REAL*4 surfPrecipTotRateUn(536,1440,2)
    REAL*4 surfPrecipLiqRateUn(536,1440,2)
    REAL*4 surfPrecipTotRateProb(536,1440,2)
    REAL*4 surfPrecipLiqRateProb(536,1440,2)
END STRUCTURE

STRUCTURE /L3CMBTX_G1_SURFPRECIPTOTRATEDIURNAL/
  INTEGER*4 count(28,72,2,3,24)
  REAL*4 mean(28,72,2,3,24)
  REAL*4 stdev(28,72,2,3,24)
END STRUCTURE

STRUCTURE /L3CMBTX_G1_PRECIPTOTLOGNW/
  INTEGER*4 count(28,72,2,16,3,3)
  REAL*4 mean(28,72,2,16,3,3)
  REAL*4 stdev(28,72,2,16,3,3)
  INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBTX_G1_PRECIPTOTDM/
  INTEGER*4 count(28,72,2,16,3,3)
  REAL*4 mean(28,72,2,16,3,3)
  REAL*4 stdev(28,72,2,16,3,3)
  INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBTX_G1_CLOUDLIQWATERCONTENT/
  INTEGER*4 count(28,72,2,16,3)
  REAL*4 mean(28,72,2,16,3)
  REAL*4 stdev(28,72,2,16,3)
  INTEGER*4 hist(28,72,2,16,3,30)
END STRUCTURE
STRUCTURE /L3CMBTX_G1_PRECIPLIQWATERCONTENT/
    INTEGER*4 count(28,72,2,16,3,3)
    REAL*4 mean(28,72,2,16,3,3)
    REAL*4 stdev(28,72,2,16,3,3)
    INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBTX_G1_PRECIPTOTWATERCONTENT/
    INTEGER*4 count(28,72,2,16,3,3)
    REAL*4 mean(28,72,2,16,3,3)
    REAL*4 stdev(28,72,2,16,3,3)
    INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBTX_G1_PRECIPLIQRATE/
    INTEGER*4 count(28,72,2,16,3,3)
    REAL*4 mean(28,72,2,16,3,3)
    REAL*4 stdev(28,72,2,16,3,3)
    INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBTX_G1_PRECIPTOTRATE/
    INTEGER*4 count(28,72,2,16,3,3)
    REAL*4 mean(28,72,2,16,3,3)
    REAL*4 stdev(28,72,2,16,3,3)
    INTEGER*4 hist(28,72,2,16,3,3,30)
END STRUCTURE

STRUCTURE /L3CMBTX_G1/
    RECORD /L3CMBTX_G1_PRECIPTOTRATE/ precipTotRate
    RECORD /L3CMBTX_G1_PRECIPLIQRATE/ precipLiqRate
    RECORD /L3CMBTX_G1_PRECIPTOTWATERCONTENT/ precipTotWaterContent
    RECORD /L3CMBTX_G1_PRECIPLIQWATERCONTENT/ precipLiqWaterContent
    RECORD /L3CMBTX_G1_CLOUDLIQWATERCONTENT/ cloudLiqWaterContent
    RECORD /L3CMBTX_G1_PRECIPTOTDM/ precipTotDm
    RECORD /L3CMBTX_G1_PRECIPTOTLOGNW/ precipTotLogNw
    INTEGER*4 precipAllObs(28,72,2,16,3)
    RECORD /L3CMBTX_G1_SURFPRECIPTOTRATEDIURNAL/ surfPrecipTotRateDiurnal
    INTEGER*4 surfPrecipTotRateDiurnalAllObs(28,72,2,3,24)
    REAL*4 surfPrecipTotRateUn(28,72,2)
    REAL*4 surfPrecipLiqRateUn(28,72,2)
    REAL*4 surfPrecipTotRateProb(28,72,2)
REAL*4 surfPrecipLiqRateProb(28,72,2)
END STRUCTURE

STRUCTURE /L3CMBTX_GRIDS/
  RECORD /L3CMBTX_G1/ G1
  RECORD /L3CMBTX_G2/ G2
END STRUCTURE

5.73 3GSMAPH4 - GSMaP Hourly

3GSMAPH, "GSMaP Hourly", provides precipitation estimates at 0.1 degrees by the
Global Satellite Mapping of Precipitation (GSMaP). GSMaP provides high-precision,
high-resolution global precipitation maps using satellite data. The PI is JAXA. The
granule size is 1 hour. The following sections describe the structure and contents of the
format.

Dimension definitions:
   nlat  1800  Number of 0.1° grid intervals of latitude from 90° S to 90° N.
   nlon  3600  Number of 0.1° grid intervals of longitude from 180° W to 180° E.
   n8     8    Number 8.

Figure 1128 shows the structure of this product. The text below describes the contents of
objects in the structure, the C Structure Header File and the Fortran Structure Header
File.

FileHeader (Metadata):  
FileHeader contains general metadata. This group appears in all data products. See
Metadata for GPM Products for details.

FileInfo (Metadata):  
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in
all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):  
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP.
See Metadata for GPM Products for details.

GSMaPInfo (Metadata):  
GSMaPInfo contains metadata required by GSMaP. Used by GSMaP products only. See
Metadata for GPM Products for details.

Grid (Grid)
GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

Latitude (4-byte float, array size: nlat x nlon):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9 Missing value

Longitude (4-byte float, array size: nlat x nlon):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

hourlyPrecipRate (4-byte float, array size: nlat x nlon):
hourlyPrecipRate indicates hourly precipitation rate at each pixel. Unit is [mm/hr]. Negative value denotes missing in observation data or no precipitation rate was retrieved within microwave algorithms. Detailed description for missing data is shown below.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.0 or positive)</td>
<td>Hourly precipitation rate [mm/hr].</td>
</tr>
<tr>
<td>-4</td>
<td>Missing due to sea ice within microwave algorithms.</td>
</tr>
</tbody>
</table>
-8 Missing due to low temperature within microwave algorithms.
-9999.9 Missing due to no observation by IR and/or microwave sensor.

**satelliteInfoFlag** (8-byte integer, array size: nlat x nlon):
satelliteInfoFlag indicates the information of all satellite/sensor which are used in estimation of precipitation rate at each pixel during one-hour time period. Data are stored in signed 8-byte integer (64-bit). Satellite and sensor name are assigned to each bit. If the flag shows value of 0, there is no satellite observation by both microwave and geo-stationary IR sensor. Missing value is defined as -99. Negative values indicates no microwave radiometer observation at that pixel. Below is a list of pixel values, bit, and corresponding instrument.

<table>
<thead>
<tr>
<th>Value</th>
<th>Bit</th>
<th>Sensor Category</th>
<th>Satellite/Sensor</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>Infrared Imager</td>
<td>NOAA/CPC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aboard Geo-stationary</td>
<td>Globally Merged IR data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>meteorological satellite</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>Microwave radiometer</td>
<td>TRMM/TMI</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(imager/sounder) aboard</td>
<td>low orbital satellite</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td></td>
<td>GPM-Core/CMI</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td></td>
<td>Megha-Tropiques/MADRAS</td>
</tr>
<tr>
<td>16</td>
<td>4</td>
<td></td>
<td>Megha-Tropiques/SAPHIR</td>
</tr>
<tr>
<td>32</td>
<td>5</td>
<td></td>
<td>ADEOS-II/AMSR</td>
</tr>
<tr>
<td>64</td>
<td>6</td>
<td></td>
<td>Aqua/AMSR-E</td>
</tr>
<tr>
<td>128</td>
<td>7</td>
<td></td>
<td>GCOM-W1/AMSR2</td>
</tr>
<tr>
<td>256</td>
<td>8</td>
<td></td>
<td>GCOM-W2/AMSR2 f/o (TBD)</td>
</tr>
<tr>
<td>512</td>
<td>9</td>
<td></td>
<td>GCOM-W3/AMSR2 f/o (TBD)</td>
</tr>
<tr>
<td>1024</td>
<td>10</td>
<td></td>
<td>DMSP-F11/SSM/I</td>
</tr>
<tr>
<td>2048</td>
<td>11</td>
<td></td>
<td>DMSP-F13/SSM/I</td>
</tr>
<tr>
<td>4096</td>
<td>12</td>
<td></td>
<td>DMSP-F14/SSM/I</td>
</tr>
<tr>
<td>8192</td>
<td>13</td>
<td></td>
<td>DMSP-F15/SSM/I</td>
</tr>
<tr>
<td>16384</td>
<td>14</td>
<td></td>
<td>DMSP-F16/SSMIS</td>
</tr>
<tr>
<td>32768</td>
<td>15</td>
<td></td>
<td>DMSP-F17/SSMIS</td>
</tr>
<tr>
<td>65536</td>
<td>16</td>
<td></td>
<td>DMSP-F18/SSMIS</td>
</tr>
<tr>
<td>131072</td>
<td>17</td>
<td></td>
<td>DMSP-F19/SSMIS</td>
</tr>
<tr>
<td>262144</td>
<td>18</td>
<td></td>
<td>DMSP-F20/SSMIS</td>
</tr>
<tr>
<td>524288</td>
<td>19</td>
<td></td>
<td>NOAA-15/AMSU-A/B</td>
</tr>
<tr>
<td>1048576</td>
<td>20</td>
<td></td>
<td>NOAA-16/AMSU-A/B</td>
</tr>
<tr>
<td>2097152</td>
<td>21</td>
<td></td>
<td>NOAA-17/AMSU-A/B</td>
</tr>
<tr>
<td>4194304</td>
<td>22</td>
<td></td>
<td>NOAA-18/AMSU-A/MHS</td>
</tr>
<tr>
<td>8388608</td>
<td>23</td>
<td></td>
<td>NOAA-19/AMSU-A/MHS</td>
</tr>
<tr>
<td>16777216</td>
<td>24</td>
<td></td>
<td>NPP/ATMS</td>
</tr>
</tbody>
</table>
33554432 25  JPSS-1/ATMS
67108864 26  MetOp-A/AMSU-A/MHS
134217728 27  MetOp-B/AMSU-A/MHS
268435456 28  MetOp-C/AMSU-A/MHS
29-63  Spare

**observationTimeFlag** (4-byte float, array size: nlat x nlon):

observationTimeFlag indicates relative time of nearest microwave radiometer (imager/sounder) observation to start time of the file at each pixel. Data are stored in 4-byte float. Value of 0 means start time of the file (HH in file name). Missing value is defined as -9999.9. Detailed description is below.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 LE X LT 1</td>
<td>If value is positive and smaller than 1, microwave radiometer observation is available at the pixel during current one-hour period. X indicates relative observation time of latest microwave radiometer, and is stored as difference from the start time of the file. For example, if UTC of the file (HH) = 01 and X = 0.2, observation time of the pixel will be 01:12 UTC.</td>
</tr>
<tr>
<td>1 LE X</td>
<td>If value is equal to or larger than 1, NO microwave radiometer observation is available at the pixel during time period of the file. X indicates relative observation time of coming microwave radiometer, and stored as differences from the start time of the file. For example, if UTC of the file (HH) = 01 and X= 2.5, coming observation time of microwave radiometer at the pixel will be 3:30 UTC.</td>
</tr>
<tr>
<td>X LT 0</td>
<td>If value is negative, NO</td>
</tr>
</tbody>
</table>
microwave radiometer observation is available at the pixel during time period of the file. X (X LT 0) indicates relative observation time of latest microwave radiometer, and stored as differences from the start time of the file. For example, if UTC of the file (HH) = 01 and X = -2.5, latest observation time of microwave radiometer at the pixel will be 22:30 UTC of previous day. X = -9999.9 No microwave observation (Missing)

**hourlyPrecipRateGC** (4-byte float, array size: nlat x nlon):
hourlyPrecipRateGC indicates hourly precipitation rate that was corrected by rain gauge data (NOAA CPC Unified Gauge-Based Analysis of Global Daily Precipitation, in daily and 0.5-degree grid) at each pixel. Data are stored in 4-byte float. Unit is [mm/hr]. Missing value is defined as -9999.9.

**gaugeQualityInfo** (2-byte integer, array size: nlat x nlon):
gaugeQualityInfo indicates the number of gauge data in original 0.5-degree pixel and daily, which was used in calculation of hourlyPrecipRateGC. Data are stored in 4-byte integer. Unit is [counts/day]. Missing value is defined as -9999.

**snowProbability** (2-byte integer, array size: nlat x nlon):
Probability of snow in percent. Data are stored in 2-byte integer. Range is 0 to 100. Missing value is defined as -9999.

**C Structure Header file:**

```c
#ifndef _TK_3GSMAPH4_H_
#define _TK_3GSMAPH4_H_

#ifndef _L3GSMAPH4_GRID_
#define _L3GSMAPH4_GRID_

typedef struct {
    float Latitude[3600][1800];
    float Longitude[3600][1800];
    float hourlyPrecipRate[3600][1800];
    long long satelliteInfoFlag[3600][1800];
}
```
float observationTimeFlag[3600][1800];
float hourlyPrecipRateGC[3600][1800];
short gaugeQualityInfo[3600][1800];
short snowProbability[3600][1800];
} L3GSMAPH4_GRID;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L3GSMAPH4_GRID/
  REAL*4 Latitude(1800,3600)
  REAL*4 Longitude(1800,3600)
  REAL*4 hourlyPrecipRate(1800,3600)
  INTEGER*8 satelliteInfoFlag(1800,3600)
  REAL*4 observationTimeFlag(1800,3600)
  REAL*4 hourlyPrecipRateGC(1800,3600)
  INTEGER*2 gaugeQualityInfo(1800,3600)
  INTEGER*2 snowProbability(1800,3600)
END STRUCTURE

5.74 3GSMAPM4 - GSMaP Monthly

3GSMAPM, "GSMaP Monthly", provides precipitation estimates at 0.1 degrees by the Global Satellite Mapping of Precipitation (GSMaP). GSMaP provides high-precision, high-resolution global precipitation maps using satellite data. The PI is JAXA. The granule size is 1 month. The following sections describe the structure and contents of the format.

Dimension definitions:

  nlat  1800  Number of 0.1° grid intervals of latitude from 90° S to 90° N.
  nlon  3600  Number of 0.1° grid intervals of longitude from 180° W to 180° E.

Figure 1129 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.
Figure 1129: Data Format Structure for 3GSMAPM4, GSMaP Monthly

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo (Metadata):**
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**GSMaPInfo (Metadata):**
GSMaPInfo contains metadata required by GSMaP. Used by GSMaP products only. See Metadata for GPM Products for details.

**Grid (Grid)**

**GridHeader (Metadata):**
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**Longitude (4-byte float, array size: nlat x nlon):**
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. Values range from -180 to 180 degrees. Special values are
defined as:
-9999.9 Missing value

**Longitude** (4-byte float, array size: nlat x nlon):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9 Missing value

**monthlyPrecipRate** (4-byte float, array size: nlat x nlon):
monthlyPrecipRate indicates monthly precipitation rate at each pixel. Unit is [mm/hr]. Negative value denotes missing in observation data or no precipitation rate was retrieved within microwave algorithms. Detailed description for missing data is shown below.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.0 or positive)</td>
<td>Monthly precipitation rate [mm/hr].</td>
</tr>
<tr>
<td>-4</td>
<td>Missing due to sea ice within microwave algorithms.</td>
</tr>
<tr>
<td>-8</td>
<td>Missing due to low temperature within microwave algorithms.</td>
</tr>
<tr>
<td>-9999.9</td>
<td>Missing due to no observation by IR and/or microwave sensor.</td>
</tr>
</tbody>
</table>

**observationNumber** (4-byte integer, array size: nlat x nlon):
obervationNumber indicates the number of observation that was used in the estimation of monthly mean precipitation rate at each pixel during one month. Data are stored in 4-byte integer. Unit is [counts/month]. Missing value is defined as -9999.

**standardDeviation** (4-byte float, array size: nlat x nlon):
standardDeviation indicates monthly standard deviation of precipitation rate at each pixel. Data are stored in 4-byte float. Unit is [mm/hr]. Missing value is defined as -9999.9.

**monthlyPrecipRateGC** (4-byte float, array size: nlat x nlon): 
monthlyPrecipRateGC indicates monthly mean precipitation rate of hourlyPrecipRateGC. Data are stored in 4-byte float. Unit is [mm/hr]. Missing value is defined as -9999.9.

**gaugeQualityInfo** (2-byte integer, array size: nlat x nlon):
gaugeQualityInfo indicates the number of gauge data in original 0.5-degree pixel, which was used in calculation of monthlyPrecipRateGC. Data are stored in 4-byte integer. Unit is [counts/month]. Missing value is defined as -9999. Special values are defined as:
-9999 Missing value

**snowProbability** (2-byte integer, array size: nlat x nlon):
Probability of snow in percent. Data are stored in 2-byte integer. Range is 0 to 100. Missing value is defined as -9999.

**C Structure Header file:**

```c
#ifndef _TK_3GSMAPM4_H_
define _TK_3GSMAPM4_H_
```
#ifndef _L3GSMAPM4_GRID_
define _L3GSMAPM4_GRID_

typedef struct {
    float Latitude[3600][1800];
    float Longitude[3600][1800];
    float monthlyPrecipRate[3600][1800];
    int observationNumber[3600][1800];
    float standardDeviation[3600][1800];
    float monthlyPrecipRateGC[3600][1800];
    short gaugeQualityInfo[3600][1800];
    short snowProbability[3600][1800];
} L3GSMAPM4_GRID;

#endif

Fortran Structure Header file:

STRUCTURE /L3GSMAPM4_GRID/
    REAL*4 Latitude(1800,3600)
    REAL*4 Longitude(1800,3600)
    REAL*4 monthlyPrecipRate(1800,3600)
    INTEGER*4 observationNumber(1800,3600)
    REAL*4 standardDeviation(1800,3600)
    REAL*4 monthlyPrecipRateGC(1800,3600)
    INTEGER*2 gaugeQualityInfo(1800,3600)
    INTEGER*2 snowProbability(1800,3600)
END STRUCTURE

5.75 3IMERGHH - IMERG 30-minute

3IMERGHH, "IMERG 30-minute", provides precipitation estimates at 0.1 degrees by the Integrated Multi-satellite Retrievals for GPM (IMERG). IMERG is intended to intercalibrate, merge, and interpolate satellite microwave precipitation estimates, together with microwave-calibrated infrared (IR) satellite estimates, and precipitation gauge analyses. The PI is Dr. George Huffman. The granule size is 30 minutes. The following sections describe the structure and contents of the format.

Dimension definitions:
nv 2 Number of time bounds.
lonv 2 Number of longitude bounds.
latv 2 Number of latitude bounds.
time var Number of times in data set.
lon 3600 Number of 0.1° grid intervals of longitude from 180° W to 180° E.
lat 1800 Number of 0.1° grid intervals of latitude from 90° S to 90° N.

Figure 1130 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.
FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

time (4-byte integer, array size: time):
Representative time of data in seconds since 1970-01-01 00:00:00 UTC. In V6 the calculation of time did not add leap seconds. The difference due to leap seconds grows with time and in 2019 is less than 30 seconds.

lon (4-byte float, array size: lon):
Longitude at the center of 0.1° grid intervals of longitude from 180° W to 180° E.

lat (4-byte float, array size: lat):
Latitude at the center of 0.1° grid intervals of latitude from 90° S to 90° N.

time_bnds (4-byte integer, array size: nv x time):
Start and stop time of the data. In V6 the calculation of time did not add leap seconds. The difference due to leap seconds grows with time and in 2019 is less than 30 seconds. Values range from 0 to 2147000000 seconds since 1970-01-01 00:00:00 UTC. Special values are defined as:
  -9999  Missing value

lon_bnds (4-byte float, array size: lonv x lon):
Longitude of the west and east edges of the grid boxes. Values range from -180 to 180 degrees_east. Special values are defined as:
  -9999.9  Missing value

lat_bnds (4-byte float, array size: latv x lat):
Latitude of the south and north edges of the grid boxes. Values range from -90 to 90 degrees_north. Special values are defined as:
  -9999.9  Missing value

precipitationCal (4-byte float, array size: lat x lon x time):
Precipitation estimate using gauge calibration over land. Values range from 0 to 1000 mm/hr. Special values are defined as:
  -9999.9  Missing value

precipitationUncal (4-byte float, array size: lat x lon x time):
Precipitation estimate with no gauge calibration. Values range from 0 to 1000 mm/hr. Special values are defined as:
  -9999.9  Missing value
randomError (4-byte float, array size: lat x lon x time):
Random error estimate of precipitation. Values range from 0 to 1000 mm/hr. Special values are defined as:
-9999.9  Missing value

HQprecipitation (4-byte float, array size: lat x lon x time):
Instantaneous microwave-only precipitation estimate covering the current 30-minute period. Values range from 0 to 1000 mm/hr. Special values are defined as:
-9999.9  Missing value

HQprecipSource (2-byte integer, array size: lat x lon x time):

HQprecipSource values are as follows:
0  = no observation
1  = TMI
2  = TCI
3  = AMSR
4  = SSMI
5  = SSMIS
6  = AMSU
7  = MHS
8  = Megha-Tropiques
9  = GMI
10  = GCI
11  = ATMS
12  = AIRS
13  = TOVS
14  = Cr1s
15  = future microwave scanner
16  = future microwave scanner
17  = future microwave scanner
18  = future microwave scanner
19  = future microwave scanner
20  = future microwave sounder
21  = future microwave sounder
22  = future microwave sounder
23  = future microwave sounder
24  = future microwave sounder

Satellite ID of the instantaneous microwave-only precipitation estimate covering the current 30-minute period. Values range from 0 to 24.

HQobservationTime (2-byte integer, array size: lat x lon x time):
Observation time (from the beginning of the current half hour) of the instantaneous microwave-only precipitation estimate covering the current 30-minute period. Values
range from 0 to 29 minutes. Special values are defined as:
  -9999  Missing value

**IRprecipitation** (4-byte float, array size: lat x lon x time):
Microwave-calibrated IR precipitation estimate covering the current 30-minute period. Values range from 0 to 1000 mm/hr. Special values are defined as:
  -9999.9  Missing value

**IRkalmanFilterWeight** (2-byte integer, array size: lat x lon x time):
IR weighting in the final precipitation estimate. The values range from 0 to 100, where 0 is no IR weighting and 100 is entirely based on IR. A value of 0 is provided as well in areas of no precipitation.

**probabilityLiquidPrecipitation** (2-byte integer, array size: lat x lon x time):
Probability of liquid precipitation. 0=definitely frozen. 100=definitely liquid. 50=equal probability frozen or liquid. This field is globally complete and provided irrespective of the presence of precipitation. Values range from 0 to 100 percent.

**precipitationQualityIndex** (4-byte float, array size: lat x lon x time):
Estimated quality of precipitationCal where 0 is worse and 100 is better. Values range from 0 to 100. Special values are defined as:
  -9999.9  Missing value

**C Structure Header file:**

```c
#ifndef _TK_3IMERGHH_H_
#define _TK_3IMERGHH_H_

#ifndef _L3IMERGHH_GRID_
#define _L3IMERGHH_GRID_

typedef struct {
  int time[1];
  float lon[3600];
  float lat[1800];
  int time_bnds[1][2];
  float lon_bnds[3600][2];
  float lat_bnds[1800][2];
  float precipitationCal[1][3600][1800];
  float precipitationUncal[1][3600][1800];
  float randomError[1][3600][1800];
  float HQprecipitation[1][3600][1800];
  short HQprecipSource[1][3600][1800];
  short HQobservationTime[1][3600][1800];
  float IRprecipitation[1][3600][1800];
  short IRkalmanFilterWeight[1][3600][1800];

```
Fortran Structure Header file:

```fortran
STRUCTURE /L3IMERGHH_GRID/
    INTEGER*4 time(1)
    REAL*4 lon(3600)
    REAL*4 lat(1800)
    INTEGER*4 time_bnds(2,1)
    REAL*4 lon_bnds(2,3600)
    REAL*4 lat_bnds(2,1800)
    REAL*4 precipitationCal(1800,3600,1)
    REAL*4 precipitationUncal(1800,3600,1)
    REAL*4 randomError(1800,3600,1)
    REAL*4 HQprecipitation(1800,3600,1)
    INTEGER*2 HQprecipSource(1800,3600,1)
    INTEGER*2 HQobservationTime(1800,3600,1)
    REAL*4 IRprecipitation(1800,3600,1)
    INTEGER*2 IRkalmanFilterWeight(1800,3600,1)
    INTEGER*2 probabilityLiquidPrecipitation(1800,3600,1)
    REAL*4 precipitationQualityIndex(1800,3600,1)
END STRUCTURE
```

5.76 3IMERGM - IMERG monthly

3IMERGM, “IMERG monthly”, provides precipitation estimates at 0.1 degrees by the Integrated Multi-satellitE Retrievals for GPM (IMERG). IMERG is intended to intercalibrate, merge, and interpolate satellite microwave precipitation estimates, together with microwave-calibrated infrared (IR) satellite estimates, and precipitation gauge analyses. The PI is Dr. George Huffman. The granule size is 1 month. The following sections describe the structure and contents of the format.

Dimension definitions:
Figure 1131: Data Format Structure for 3IMERGM, IMERG monthly

nv 2 Number of time bounds.
lonv 2 Number of longitude bounds.
lav 2 Number of latitude bounds.
time var Number of times in data set.
lon 3600 Number of 0.1° grid intervals of longitude from 180° W to 180° E.
lav 1800 Number of 0.1° grid intervals of latitude from 90° S to 90° N.

Figure 1131 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

Grid (Grid)
GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

time (4-byte integer, array size: time):
Representative time of data in seconds since 1970-01-01 00:00:00 UTC. In V6 the calculation of time did not add leap seconds. The difference due to leap seconds grows with time and in 2019 is less than 30 seconds.

lon (4-byte float, array size: lon):
Longitude at the center of 0.1° grid intervals of longitude from 180° W to 180° E.

lat (4-byte float, array size: lat):
Latitude at the center of 0.1° grid intervals of latitude from 90° S to 90° N.

time_bnds (4-byte integer, array size: nv x time):
Start and stop time of the data. In V6 the calculation of time did not add leap seconds. The difference due to leap seconds grows with time and in 2019 is less than 30 seconds. Values range from 0 to 2147000000 seconds since 1970-01-01 00:00:00 UTC. Special values are defined as:
-9999 Missing value

lon_bnds (4-byte float, array size: lonv x lon):
Longitude of the west and east edges of the grid boxes. Values range from -180 to 180 degrees_east. Special values are defined as:
-9999.9 Missing value

lat_bnds (4-byte float, array size: latv x lat):
Latitude of the south and north edges of the grid boxes. Values range from -90 to 90 degrees_north. Special values are defined as:
-9999.9 Missing value

precipitation (4-byte float, array size: lat x lon x time):
Precipitation estimate using gauge calibration over land. Values range from 0 to 1000 mm/hr. Special values are defined as:
-9999 Missing value

randomError (4-byte float, array size: lat x lon x time):
Random error estimate of precipitation. Values range from 0 to 1000 mm/hr. Special values are defined as:
-9999.9 Missing value

gaugeRelativeWeighting (2-byte integer, array size: lat x lon x time):
Surface gauge weighting in the final precipitation estimate. The values range from 0 to 100, where 0 is no gauge weighting and 100 is entirely based on gauge. Values range from 0 to 100 percent. Special values are defined as:
-9999 Missing value

probabilityLiquidPrecipitation (2-byte integer, array size: lat x lon x time):
Probability of liquid precipitation. 0=definitely frozen. 100=definitely liquid. 50=equal
probability frozen or liquid. This field is globally complete and provided irrespective of
the presence of precipitation. Values range from 0 to 100 percent.

precipitationQualityIndex (4-byte float, array size: lat x lon x time):
    Estimated quality of precipitation where 0 is worse and 100 is better. Values range
    from 0 to 100. Special values are defined as:
    -9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_3IMERGM_H_
#define _TK_3IMERGM_H_

#ifndef _L3IMERGM_GRID_
#define _L3IMERGM_GRID_

typedef struct {
    int time[1];
    float lon[3600];
    float lat[1800];
    int time_bnds[1][2];
    float lon_bnds[3600][2];
    float lat_bnds[1800][2];
    float precipitation[1][3600][1800];
    float randomError[1][3600][1800];
    short gaugeRelativeWeighting[1][3600][1800];
    short probabilityLiquidPrecipitation[1][3600][1800];
    float precipitationQualityIndex[1][3600][1800];
} L3IMERGM_GRID;

#endif
#endif
```

```fortran
STRUCTURE /L3IMERGM_GRID/
    INTEGER*4 time(1)
    REAL*4 lon(3600)
    REAL*4 lat(1800)
    INTEGER*4 time_bnds(2,1)
    REAL*4 lon_bnds(2,3600)
    REAL*4 lat_bnds(2,1800)
    REAL*4 precipitation(1800,3600,1)
```
2HSLH, "Spectral Latent Heating," produces latent heating, Q1-QR, and Q2 profiles from DPR rain. The PI is Dr. Takayabu and the Co-PI is Dr. Shige. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:
- **nscan**: Number of scans in the granule.
- **nray**: Number of angle bins in each scan.
- **nlayer**: Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1132 through Figure 1134 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.
File

Swath

SwathHeader

ScanTime 19 bytes

Group: nscan

Latitude 4 bytes

Array: nray x nscan

Longitude 4 bytes

Array: nray x nscan

latentHeating 4 bytes

Array: nlayer x nray x nscan

Q1minusQR 4 bytes

Array: nlayer x nray x nscan

Q2 4 bytes

Array: nlayer x nray x nscan

rainTypeSLH 2 bytes

Array: nray x nscan

stormTopHeight 2 bytes

Array: nray x nscan

meltLayerHeight 2 bytes

Array: nray x nscan

nearSurfLevel 2 bytes

Array: nray x nscan

topoLevel 2 bytes

Array: nray x nscan

climMeltLevel 2 bytes

Array: nray x nscan

climFreezLevel 2 bytes

Array: nray x nscan

nearSurfacePrecipRate 4 bytes

Array: nray x nscan

precipRateMeltLevel 4 bytes

Array: nray x nscan

precipRateClimFreezLevel 4 bytes

Array: nray x nscan

continued on next figure

Figure 1132: Data Format Structure for 2HSLH, Spectral Latent Heating
continued from last figure

Figure 1133: Data Format Structure for 2HSLH, Spectral Latent Heating

Figure 1134: Data Format Structure for 2HSLH, ScanTime
Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value
**Latitude** (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
-9999.9  Missing value

**Longitude** (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**latentHeating** (4-byte float, array size: nlayer x nray x nscan):
Latent Heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**Q1minusQR** (4-byte float, array size: nlayer x nray x nscan):
Q1 - QR. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**Q2** (4-byte float, array size: nlayer x nray x nscan):
Apparent moisture sink. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**rainTypeSLH** (2-byte integer, array size: nray x nscan):
Rain type decided by SLH. SLH decides if the FOV is tropical or mid-latitude based on the monthly precipitation regime database. The decision is not based on fixed latitude. Values are as follows:

0: No precipitation - all latitudes
1: Convective - tropical
2: Shallow stratiform - tropical
3: Deep stratiform - tropical
4: Deep stratiform with low melting level - tropical
5: Intermediary - tropical
6: Other - tropical
110: Convective - mid-latitude
121: Shallow stratiform - mid-latitude
122: Deep stratiform, downward decreasing - mid-latitude
123: Deep stratiform, downward increasing - mid-latitude
124: Deep stratiform, subzero - mid-latitude
160: Other - mid-latitude
900: Tibet, winter mid-lat etc. (masked)
910: Suspicious extreme (masked)
-9999: Missing value
stormTopHeight (2-byte integer, array size: nray x nscan):
Height of storm top. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

meltLayerHeight (2-byte integer, array size: nray x nscan):
Height of melting layer. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

earSurfLevel (2-byte integer, array size: nray x nscan):
Level of near surface rain. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

topoLevel (2-byte integer, array size: nray x nscan):
Level of topography. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

climMeltLevel (2-byte integer, array size: nray x nscan):
Climatological melting level. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

climFreezLevel (2-byte integer, array size: nray x nscan):
Climatological freezing level. Values range from 0 to 32000 m. Special values are defined as:
-9999 Missing value

nearSurfacePrecipRate (4-byte float, array size: nray x nscan):
Precipitation rate at the near surface. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateMeltLevel (4-byte float, array size: nray x nscan):
Precipitation rate at the melting level. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9 Missing value

precipRateClimFreezLevel (4-byte float, array size: nray x nscan):
Precipitation rate at the freezing level. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9 Missing value

rainType2ADPR (2-byte integer, array size: nray x nscan):
Rain Type from 2ADPR. Special values are defined as:
-9999 Missing value

method (2-byte integer, array size: nray x nscan):
Method from 2ADPR. Special values are defined as:
-9999 Missing value

C Structure Header file:
#ifndef _TK_2HSLH_H_
define _TK_2HSLH_H_

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#endif

#ifndef _L2HSLH_SWATH_
define _L2HSLH_SWATH_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    float latentHeating[49][80];
    float Q1minusQR[49][80];
    float Q2[49][80];
    short rainTypeSLH[49];
    short stormTopHeight[49];
    short meltLayerHeight[49];
    short nearSurfLevel[49];
    short topoLevel[49];
    short climMeltLevel[49];
    short climFreezLevel[49];
    float nearSurfacePrecipRate[49];
    float precipRateMeltLevel[49];
    float precipRateClimFreezLevel[49];
    short rainType2ADPR[49];
    short method[49];
} L2HSLH_SWATH;
#endif
#endif

Fortran Structure Header file:

```
STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2HSLH_SWATH/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    REAL*4 latentHeating(80,49)
    REAL*4 Q1minusQR(80,49)
    REAL*4 Q2(80,49)
    INTEGER*2 rainTypeSLH(49)
    INTEGER*2 stormTopHeight(49)
    INTEGER*2 meltLayerHeight(49)
    INTEGER*2 nearSurfLevel(49)
    INTEGER*2 topoLevel(49)
    INTEGER*2 climMeltLevel(49)
    INTEGER*2 climFreezLevel(49)
    REAL*4 nearSurfacePrecipRate(49)
    REAL*4 precipRateMeltLevel(49)
    REAL*4 precipRateClimFreezLevel(49)
    INTEGER*2 rainType2ADPR(49)
    INTEGER*2 method(49)
END STRUCTURE
```
5.78 3GSLH - Gridded Orbital Spectral Latent Heating

3GSLH, "Gridded Orbital Spectral Latent Heating", produces 0.5° x 0.5° latent heating, Q1-QR, and Q2 profiles from DPR rain. The PI is Dr. Takayabu and the Co-PI is Dr. Shige. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:

- nlat 268 Number of 0.5° grid intervals of latitude from 67°N to 67°S.
- nlon 720 Number of 0.5° grid intervals of longitude from 180°W to 180°E.
- nlayer 80 Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1135 through Figure 1137 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

allLHUnCndMean (4-byte float, array size: nlat x nlon x nlayer):
Latent heating: all pixel unconditional mean. Values range from -400 to 400 K/hr. Special
Figure 1135: Data Format Structure for 3GSLH, Gridded Orbital Spectral Latent Heating
continued from last figure

- **otherQ2CndMean**: 4 bytes
  - Array: nlat x nlon x nlayer
- **allPix**: 2 bytes
  - Array: nlat x nlon x nlayer
- **precipPix**: 2 bytes
  - Array: nlat x nlon x nlayer
- **convPix**: 2 bytes
  - Array: nlat x nlon x nlayer
- **dpstrPix**: 2 bytes
  - Array: nlat x nlon x nlayer
- **shstrPix**: 2 bytes
  - Array: nlat x nlon x nlayer
- **otherPix**: 2 bytes
  - Array: nlat x nlon x nlayer

**GridTime**

- **Year**: 2 bytes
  - Array: nlat x nlon
- **Month**: 1 byte
  - Array: nlat x nlon
- **DayOfMonth**: 1 byte
  - Array: nlat x nlon
- **Hour**: 1 byte
  - Array: nlat x nlon
- **Minute**: 1 byte
  - Array: nlat x nlon
- **Second**: 1 byte
  - Array: nlat x nlon
- **MilliSecond**: 2 bytes
  - Array: nlat x nlon
- **DayOfYear**: 2 bytes
  - Array: nlat x nlon

**Figure 1136**: Data Format Structure for 3GSLH, Gridded Orbital Spectral Latent Heating

**Figure 1137**: Data Format Structure for 3GSLH, GridTime
values are defined as:
-9999.9 Missing value

**allQ1UnCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1-QR: all pixel unconditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**allQ2UnCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2: all pixel unconditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**allLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating all pixel mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**convLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**dpstrLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating deep-stratiform and shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**shstrLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating shallow conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**otherLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating other conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**allQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR all pixel mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**convQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**dpstrQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR deep-stratiform and shallow-stratiform conditional mean. Values range from
-400 to 400 K/hr. Special values are defined as:
  -9999.9  Missing value

**shstrQ1RCndMean** *(4-byte float, array size: nlat x nlon x nlayer):*  
Q1 - QR shallow conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9  Missing value

**otherQ1RCndMean** *(4-byte float, array size: nlat x nlon x nlayer):*  
Q1 - QR other conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9  Missing value

**allQ2CndMean** *(4-byte float, array size: nlat x nlon x nlayer):*  
Q2 all pixel mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9  Missing value

**convQ2CndMean** *(4-byte float, array size: nlat x nlon x nlayer):*  
Q2 convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9  Missing value

**dpstrQ2CndMean** *(4-byte float, array size: nlat x nlon x nlayer):*  
Q2 deep-stratiform and shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9  Missing value

**shstrQ2CndMean** *(4-byte float, array size: nlat x nlon x nlayer):*  
Q2 shallow conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9  Missing value

**otherQ2CndMean** *(4-byte float, array size: nlat x nlon x nlayer):*  
Q2 other conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9  Missing value

**allPix** *(2-byte integer, array size: nlat x nlon x nlayer):*  
All pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special values are defined as:
  -9999  Missing value

**precipPix** *(2-byte integer, array size: nlat x nlon x nlayer):*  
The number of precipitating pixels in the 0.5° x 0.5° box. (= convPix + dpstrPix + shstrPix + otherPix) Values range from 0 to 500000. Special values are defined as:
  -9999  Missing value

**convPix** *(2-byte integer, array size: nlat x nlon x nlayer):*  
Convective pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special
values are defined as:
-9999 Missing value

**dpstrPix** (2-byte integer, array size: nlat x nlon x nlayer):
Deep-stratiform and shallow-stratiform pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

**shstrPix** (2-byte integer, array size: nlat x nlon x nlayer):
Shallow pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

**otherPix** (2-byte integer, array size: nlat x nlon x nlayer):
Other pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

**GridTime** (Group)
A UTC time associated with the grid box.

**Year** (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MilliSecond** (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value
**DayOfYear** (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

**C Structure Header file:**

```c
#ifndef _TK_3GSLH_H_
#define _TK_3GSLH_H_

#ifndef _L3GSLH_GRIDTIME_
#define _L3GSLH_GRIDTIME_

typedef struct {
    short Year[720][268];
    signed char Month[720][268];
    signed char DayOfMonth[720][268];
    signed char Hour[720][268];
    signed char Minute[720][268];
    signed char Second[720][268];
    short MilliSecond[720][268];
    short DayOfYear[720][268];
} L3GSLH_GRIDTIME;
#endif
#endif
#ifndef _L3GSLH_GRID_
#define _L3GSLH_GRID_

typedef struct {
    float allLHUnCndMean[80][720][268];
    float allQ1RUnCndMean[80][720][268];
    float allQ2UnCndMean[80][720][268];
    float allLHCndMean[80][720][268];
    float convLHCndMean[80][720][268];
    float dpstrLHCndMean[80][720][268];
    float shstrLHCndMean[80][720][268];
    float otherLHCndMean[80][720][268];
    float allQ1RCndMean[80][720][268];
    float convQ1RCndMean[80][720][268];
    float dpstrQ1RCndMean[80][720][268];
    float shstrQ1RCndMean[80][720][268];
    float otherQ1RCndMean[80][720][268];
    float allQ2CndMean[80][720][268];
    float convQ2CndMean[80][720][268];
```

```
float dpstrQ2CndMean[80][720][268];
float shstrQ2CndMean[80][720][268];
float otherQ2CndMean[80][720][268];
short allPix[80][720][268];
short precipPix[80][720][268];
short convPix[80][720][268];
short dpstrPix[80][720][268];
short shstrPix[80][720][268];
short otherPix[80][720][268];
L3GSLH_GRIDTIME GridTime;
} L3GSLH_GRID;

#endif
#endif

Fortran Structure Header file:

STRUCTURE /L3GSLH_GRIDTIME/
   INTEGER*2 Year(268,720)
   BYTE Month(268,720)
   BYTE DayOfMonth(268,720)
   BYTE Hour(268,720)
   BYTE Minute(268,720)
   BYTE Second(268,720)
   INTEGER*2 MilliSecond(268,720)
   INTEGER*2 DayOfYear(268,720)
END STRUCTURE

STRUCTURE /L3GSLH_GRID/
   REAL*4 allLHUnCndMean(268,720,80)
   REAL*4 allQ1RUnCndMean(268,720,80)
   REAL*4 allQ2UnCndMean(268,720,80)
   REAL*4 allLHCndMean(268,720,80)
   REAL*4 convLHCndMean(268,720,80)
   REAL*4 dpstrLHCndMean(268,720,80)
   REAL*4 shstrLHCndMean(268,720,80)
   REAL*4 otherLHCndMean(268,720,80)
   REAL*4 allQ1RCndMean(268,720,80)
   REAL*4 convQ1RCndMean(268,720,80)
   REAL*4 dpstrQ1RCndMean(268,720,80)
   REAL*4 shstrQ1RCndMean(268,720,80)
   REAL*4 otherQ1RCndMean(268,720,80)
REAL*4  allQ2CndMean(268,720,80)
REAL*4  convQ2CndMean(268,720,80)
REAL*4  dpstrQ2CndMean(268,720,80)
REAL*4  shstrQ2CndMean(268,720,80)
REAL*4  otherQ2CndMean(268,720,80)
INTEGER*2 allPix(268,720,80)
INTEGER*2 precipPix(268,720,80)
INTEGER*2 convPix(268,720,80)
INTEGER*2 dpstrPix(268,720,80)
INTEGER*2 shstrPix(268,720,80)
INTEGER*2 otherPix(268,720,80)
RECORD /L3GSLH_GRIDTIME/ GridTime
END STRUCTURE

5.79 3HSLH - Monthly Spectral Latent Heating

3HSLH, "Monthly Spectral Latent Heating", produces 0.5° x 0.5° latent heating, Q1-QR, and Q2 profiles from DPR rain. The PI is Dr. Takayabu and the Co-PI is Dr. Shige. The granule size is one month. The following sections describe the structure and contents of the format.

Dimension definitions:
  nlat   268 Number of 0.5° grid intervals of latitude from 67°S to 67°N.
  nlon   720 Number of 0.5° grid intervals of longitude from 180°W to 180°E.
  nlayer 80 Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ...
              ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1138 through Figure 1140 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputFileNames (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

InputAlgorithmVersions (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

InputGenerationDateTimes (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.
Figure 1138: Data Format Structure for 3HSLH, Monthly Spectral Latent Heating
continued from last figure

Figure 1139: Data Format Structure for 3HSLH, Monthly Spectral Latent Heating
continued from last figure

Figure 1140: Data Format Structure for 3HSLH, Monthly Spectral Latent Heating

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

LHUnCndMean (4-byte float, array size: nlat x nlon x nlayer):
Latent heating unconditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

LHUnCndStdv (4-byte float, array size: nlat x nlon x nlayer):
Latent heating unconditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

Q1RUncndMean (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR unconditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

Q1RUncndStdv (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR unconditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

Q2unCndMean (4-byte float, array size: nlat x nlon x nlayer):
Q2 unconditional mean. Values range from -400 to 400 K/hr. Special values are defined
as:

-9999.9  Missing value

**Q2UnCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 unconditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**LHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**LHCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**convLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**convLHCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating convective conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**dpstrLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating deep-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**dpstrLHCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating deep-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**shstrLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**shstrLHCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating shallow-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**otherLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating other conditional mean. Values range from -400 to 400 K/hr. Special
values are defined as:
-9999.9 Missing value

**otherLHCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating other conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**Q1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**Q1RCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**convQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**convQ1RCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR convective conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**dpstrQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR deep-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**dpstrQ1RCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR deep-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**shstrQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**shstrQ1RCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR shallow-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**otherQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR other conditional mean. Values range from -400 to 400 K/hr. Special values are
5.79  3HSLH - Monthly Spectral Latent Heating

-9999.9 Missing value

**otherQ1RCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR other conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**Q2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**Q2CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**convQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**convQ2CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 convective conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**dpstrQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 deep-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**dpstrQ2CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 deep-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**shstrQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**shstrQ2CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 shallow-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**otherQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 other conditional mean. Values range from -400 to 400 K/hr. Special values are
defined as:
-9999.9 Missing value

otherQ2CondStdv (4-byte float, array size: nlat x nlon x nlayer):
Q2 other conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

allPix (4-byte float, array size: nlat x nlon x nlayer):
All pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

precipPix (4-byte float, array size: nlat x nlon x nlayer):
The number of precipitating pixels. (= convPix + dpstrPix + shstrPix + otherPix) Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

convPix (4-byte float, array size: nlat x nlon x nlayer):
Convective pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

dpstrPix (4-byte float, array size: nlat x nlon x nlayer):
Deep-stratiform pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

shstrPix (4-byte float, array size: nlat x nlon x nlayer):
Shallow-stratiform pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

otherPix (4-byte float, array size: nlat x nlon x nlayer):
Other pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_3HSLH_H_
define _TK_3HSLH_H_
#endif

#ifndef _L3HSLH_GRID_
define _L3HSLH_GRID_
#endif

typedef struct {
    float LHUnCndMean[80][720][268];
    float LHUnCndStdv[80][720][268];
    float Q1RUnCndMean[80][720][268];
```
float Q1RUnCndStdv[80][720][268];
float Q2UnCndMean[80][720][268];
float Q2UnCndStdv[80][720][268];
float LHCndMean[80][720][268];
float LHCndStdv[80][720][268];
float convLHCndMean[80][720][268];
float convLHCndStdv[80][720][268];
float dpstrLHCndMean[80][720][268];
float dpstrLHCndStdv[80][720][268];
float shstrLHCndMean[80][720][268];
float shstrLHCndStdv[80][720][268];
float otherLHCndMean[80][720][268];
float otherLHCndStdv[80][720][268];
float Q1RCndMean[80][720][268];
float Q1RCndStdv[80][720][268];
float convQ1RCndMean[80][720][268];
float convQ1RCndStdv[80][720][268];
float dpstrQ1RCndMean[80][720][268];
float dpstrQ1RCndStdv[80][720][268];
float shstrQ1RCndMean[80][720][268];
float shstrQ1RCndStdv[80][720][268];
float otherQ1RCndMean[80][720][268];
float otherQ1RCndStdv[80][720][268];
float Q2CndMean[80][720][268];
float Q2CndStdv[80][720][268];
float convQ2CndMean[80][720][268];
float convQ2CndStdv[80][720][268];
float dpstrQ2CndMean[80][720][268];
float dpstrQ2CndStdv[80][720][268];
float shstrQ2CndMean[80][720][268];
float shstrQ2CndStdv[80][720][268];
float otherQ2CndMean[80][720][268];
float otherQ2CndStdv[80][720][268];
float allPix[80][720][268];
float precipPix[80][720][268];
float convPix[80][720][268];
float dpstrPix[80][720][268];
float shstrPix[80][720][268];
float otherPix[80][720][268];
} L3HSLH_GRID;

#endif
#endif

**Fortran Structure Header file:**

```fortran
STRUCTURE /L3HSLH_GRID/
    REAL*4 LHUnCndMean(268,720,80)
    REAL*4 LHUnCndStdv(268,720,80)
    REAL*4 Q1RUnCndMean(268,720,80)
    REAL*4 Q1RUnCndStdv(268,720,80)
    REAL*4 Q2UnCndMean(268,720,80)
    REAL*4 Q2UnCndStdv(268,720,80)
    REAL*4 LHConvMean(268,720,80)
    REAL*4 LHConvStdv(268,720,80)
    REAL*4 Q1RConvMean(268,720,80)
    REAL*4 Q1RConvStdv(268,720,80)
    REAL*4 Q2ConvMean(268,720,80)
    REAL*4 Q2ConvStdv(268,720,80)
    REAL*4 LHShstrMean(268,720,80)
    REAL*4 LHShstrStdv(268,720,80)
    REAL*4 Q1RShstrMean(268,720,80)
    REAL*4 Q1RShstrStdv(268,720,80)
    REAL*4 Q2ShstrMean(268,720,80)
    REAL*4 Q2ShstrStdv(268,720,80)
    REAL*4 LHOtherMean(268,720,80)
    REAL*4 LHOtherStdv(268,720,80)
    REAL*4 Q1ROtherMean(268,720,80)
    REAL*4 Q1ROtherStdv(268,720,80)
    REAL*4 Q2OtherMean(268,720,80)
    REAL*4 Q2OtherStdv(268,720,80)
    REAL*4 allPix(268,720,80)
    REAL*4 precipPix(268,720,80)
```

5.80 2HSLHT - Spectral Latent Heating

2HSLHT, "Spectral Latent Heating," produces latent heating, Q1-QR, and Q2 profiles from PR rain. The PI is Dr. Takayabu and the Co-PI is Dr. Shige. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:

- **nscan**: Number of scans in the granule.
- **nray**: Number of angle bins in each scan.
- **nlayer**: Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1141 through Figure 1143 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**AlgorithmRuntimeInfo** (Metadata):
AlgorithmRuntimeInfo contains text runtime information written by the algorithm. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.
Figure 1141: Data Format Structure for 2HSLHT, Spectral Latent Heating
continued from last figure

Swath

- rainType2APR: 2 bytes, Array: nray x nscan
- method: 2 bytes, Array: nray x nscan

Figure 1142: Data Format Structure for 2HSLHT, Spectral Latent Heating

ScanTime

- Year: 2 bytes, Array: nscan
- Month: 1 byte, Array: nscan
- DayOfMonth: 1 byte, Array: nscan
- Hour: 1 byte, Array: nscan
- Minute: 1 byte, Array: nscan
- Second: 1 byte, Array: nscan
- MilliSecond: 2 bytes, Array: nscan
- DayOfYear: 2 bytes, Array: nscan
- SecondOfDay: 8 bytes, Array: nscan

Figure 1143: Data Format Structure for 2HSLHT, ScanTime
Swath (Swath)

SwathHeader (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group)
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

Month (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

DayOfMonth (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

Hour (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

Minute (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

Second (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

MilliSecond (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value

DayOfYear (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999 Missing value

SecondOfDay (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9 Missing value
**Latitude** (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are defined as:
   -9999.9  Missing value

**Longitude** (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
   -9999.9  Missing value

**latentHeating** (4-byte float, array size: nlayer x nray x nscan):
Latent Heating. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**Q1minusQR** (4-byte float, array size: nlayer x nray x nscan):
Q1 - QR. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**Q2** (4-byte float, array size: nlayer x nray x nscan):
Apparent moisture sink. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**rainTypeSLH** (2-byte integer, array size: nray x nscan):
Rain type decided by SLH. SLH decides if the FOV is tropical or mid-latitude based on the monthly precipitation regime database. The decision is not based on fixed latitude. Values are as follows:

0: No precipitation - all latitudes
1: Convective - tropical
2: Shallow stratiform - tropical
3: Deep stratiform - tropical
4: Deep stratiform with low melting level - tropical
5: Intermediary - tropical
6: Other - tropical
110: Convective - mid-latitude
121: Shallow stratiform - mid-latitude
122: Deep stratiform, downward decreasing - mid-latitude
123: Deep stratiform, downward increasing - mid-latitude
124: Deep stratiform, subzero - mid-latitude
160: Other - mid-latitude
900: Tibet, winter mid-lat etc. (masked)
910: Suspicious extreme (masked)
-9999: Missing value
stormTopHeight (2-byte integer, array size: nray x nscan):
Height of storm top. Values range from 0 to 32000 m. Special values are defined as:
-9999  Missing value

meltLayerHeight (2-byte integer, array size: nray x nscan):
Height of melting layer. Values range from 0 to 32000 m. Special values are defined as:
-9999  Missing value

nearSurfLevel (2-byte integer, array size: nray x nscan):
Level of near surface rain. Values range from 0 to 32000 m. Special values are defined as:
-9999  Missing value

topoLevel (2-byte integer, array size: nray x nscan):
Level of topography. Values range from 0 to 32000 m. Special values are defined as:
-9999  Missing value

climMeltLevel (2-byte integer, array size: nray x nscan):
Climatological melting level. Values range from 0 to 32000 m. Special values are defined as:
-9999  Missing value

climFreezLevel (2-byte integer, array size: nray x nscan):
Climatological freezing level. Values range from 0 to 32000 m. Special values are defined as:
-9999  Missing value

nearSurfacePrecipRate (4-byte float, array size: nray x nscan):
Precipitation rate at the near surface. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9  Missing value

precipRateMeltLevel (4-byte float, array size: nray x nscan):
Precipitation rate at the melting level. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9  Missing value

precipRateClimFreezLevel (4-byte float, array size: nray x nscan):
Precipitation rate at the freezing level. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9  Missing value

rainType2APR (2-byte integer, array size: nray x nscan):
Rain Type from 2APR. Special values are defined as:
-9999  Missing value

method (2-byte integer, array size: nray x nscan):
Method from 2APR. Special values are defined as:
-9999  Missing value

C Structure Header file:
#ifndef _TK_2HSLHT_H_
#define _TK_2HSLHT_H_

#ifndef _SCANTIME_
#define _SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
    double SecondOfDay;
} SCANTIME;
#endif

#endif

#ifndef _L2HSLHT_SWATH_
#define _L2HSLHT_SWATH_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    float latentHeating[49][80];
    float Q1minusQR[49][80];
    float Q2[49][80];
    short rainTypeSLH[49];
    short stormTopHeight[49];
    short meltLayerHeight[49];
    short nearSurfLevel[49];
    short topoLevel[49];
    short climMeltLevel[49];
    short climFreezLevel[49];
    float nearSurfacePrecipRate[49];
    float precipRateMeltLevel[49];
    float precipRateClimFreezLevel[49];
    short rainType2APR[49];
    short method[49];
}
} L2HSLHT_SWATH;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /SCANTIME/
   INTEGER*2 Year
   BYTE Month
   BYTE DayOfMonth
   BYTE Hour
   BYTE Minute
   BYTE Second
   INTEGER*2 MilliSecond
   INTEGER*2 DayOfYear
   REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2HSLHT_SWATH/
   RECORD /SCANTIME/ ScanTime
   REAL*4 Latitude(49)
   REAL*4 Longitude(49)
   REAL*4 latentHeating(80,49)
   REAL*4 Q1minusQR(80,49)
   REAL*4 Q2(80,49)
   INTEGER*2 rainTypeSLH(49)
   INTEGER*2 stormTopHeight(49)
   INTEGER*2 meltLayerHeight(49)
   INTEGER*2 nearSurfLevel(49)
   INTEGER*2 topoLevel(49)
   INTEGER*2 climMeltLevel(49)
   INTEGER*2 climFreezLevel(49)
   REAL*4 nearSurfacePrecipRate(49)
   REAL*4 precipRateMeltLevel(49)
   REAL*4 precipRateClimFreezLevel(49)
   INTEGER*2 rainType2APR(49)
   INTEGER*2 method(49)
END STRUCTURE
5.81 3GSLHT - Gridded Orbital Spectral Latent Heating

3GSLHT, "Gridded Orbital Spectral Latent Heating", produces 0.5° x 0.5° latent heating, Q1-QR, and Q2 profiles from PR rain. The PI is Dr. Takayabu and the Co-PI is Dr. Shige. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:
- nlat 268 Number of 0.5° grid intervals of latitude from 67°S to 67°N.
- nlon 720 Number of 0.5° grid intervals of longitude from 180°W to 180°E.
- nlayer 80 Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1144 through Figure 1146 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo** (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**Grid** (Grid)

**GridHeader** (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**allLHUnCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating: all pixel unconditional mean. Values range from -400 to 400 K/hr. Special
Figure 1144: Data Format Structure for 3GSLHT, Gridded Orbital Spectral Latent Heating
continued from last figure

**Grid**

- `otherQ2CndMean` 4 bytes Array: nlat x nlon x nlayer
- `allPix` 2 bytes Array: nlat x nlon x nlayer
- `precipPix` 2 bytes Array: nlat x nlon x nlayer
- `convPix` 2 bytes Array: nlat x nlon x nlayer
- `dpstrPix` 2 bytes Array: nlat x nlon x nlayer
- `shstrPix` 2 bytes Array: nlat x nlon x nlayer
- `otherPix` 2 bytes Array: nlat x nlon x nlayer

**GridTime**

- `Year` 2 bytes Array: nlat x nlon
- `Month` 1 byte Array: nlat x nlon
- `DayOfMonth` 1 byte Array: nlat x nlon
- `Hour` 1 byte Array: nlat x nlon
- `Minute` 1 byte Array: nlat x nlon
- `Second` 1 byte Array: nlat x nlon
- `MilliSecond` 2 bytes Array: nlat x nlon
- `DayOfYear` 2 bytes Array: nlat x nlon

Figure 1145: Data Format Structure for 3GSLHT, Gridded Orbital Spectral Latent Heating

Figure 1146: Data Format Structure for 3GSLHT, GridTime
values are defined as:
   -9999.9  Missing value

**allQ1UnCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1-QR: all pixel unconditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**allQ2UnCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2: all pixel unconditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**allLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating all pixel mean. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**convLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**dpstrLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating deep-stratiform and shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**shstrLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating shallow conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**otherLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating other conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**allQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR all pixel mean. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**convQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
   -9999.9  Missing value

**dpstrQ1RCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR deep-stratiform and shallow-stratiform conditional mean. Values range from
-400 to 400 K/hr. Special values are defined as:
  -9999.9 Missing value

**shstrQ1RcndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR shallow conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9 Missing value

**otherQ1RcndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR other conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9 Missing value

**allQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 all pixel mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9 Missing value

**convQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9 Missing value

**dpstrQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 deep-stratiform and shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9 Missing value

**shstrQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 shallow conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9 Missing value

**otherQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 other conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
  -9999.9 Missing value

**allPix** (2-byte integer, array size: nlat x nlon x nlayer):
All pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special values are defined as:
  -9999 Missing value

**precipPix** (2-byte integer, array size: nlat x nlon x nlayer):
The number of precipitating pixels in the 0.5° x 0.5° box. (= convPix + dpstrPix + shstrPix + otherPix) Values range from 0 to 500000. Special values are defined as:
  -9999 Missing value

**convPix** (2-byte integer, array size: nlat x nlon x nlayer):
Convective pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special
values are defined as:
-9999 Missing value

dpstrPix (2-byte integer, array size: nlat x nlon x nlayer):
Deep-stratiform and shallow-stratiform pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

shstrPix (2-byte integer, array size: nlat x nlon x nlayer):
Shallow pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

otherPix (2-byte integer, array size: nlat x nlon x nlayer):
Other pixel counts in the 0.5° x 0.5° box. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

**GridTime** (Group)
A UTC time associated with the grid box.

**Year** (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value

**Second** (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99 Missing value

**MillisSecond** (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999 Missing value
DayOfYear (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

C Structure Header file:

```c
#ifndef _TK_3GSLHT_H_
#define _TK_3GSLHT_H_

#ifndef _L3GSLHT_GRIDTIME_
#define _L3GSLHT_GRIDTIME_

typedef struct {
    short Year[720][268];
    signed char Month[720][268];
    signed char DayOfMonth[720][268];
    signed char Hour[720][268];
    signed char Minute[720][268];
    signed char Second[720][268];
    short MilliSecond[720][268];
    short DayOfYear[720][268];
} L3GSLHT_GRIDTIME;
#endif
define _L3GSLHT_GRIDTIME_
#endif

```

```c
#include <math.h>
#include <stdlib.h>

float allLHUnCndMean[80][720][268];
float allQ1RUnCndMean[80][720][268];
float allQ2UncCndMean[80][720][268];
float allQ2UncCndMean[80][720][268];
```

float dpstrQ2CndMean[80][720][268];
float shstrQ2CndMean[80][720][268];
float otherQ2CndMean[80][720][268];
short allPix[80][720][268];
short precipPix[80][720][268];
short convPix[80][720][268];
short dpstrPix[80][720][268];
short shstrPix[80][720][268];
short otherPix[80][720][268];
L3GSLHT_GRIDTIME GridTime;
} L3GSLHT_GRID;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L3GSLHT_GRIDTIME/
  INTEGER*2 Year(268,720)
  BYTE Month(268,720)
  BYTE DayOfMonth(268,720)
  BYTE Hour(268,720)
  BYTE Minute(268,720)
  BYTE Second(268,720)
  INTEGER*2 MilliSecond(268,720)
  INTEGER*2 DayOfYear(268,720)
END STRUCTURE

STRUCTURE /L3GSLHT_GRID/
  REAL*4 allLHUnCndMean(268,720,80)
  REAL*4 allQ1RUnCndMean(268,720,80)
  REAL*4 allQ2UnCndMean(268,720,80)
  REAL*4 allLHCndMean(268,720,80)
  REAL*4 convLHCndMean(268,720,80)
  REAL*4 dpstrLHCndMean(268,720,80)
  REAL*4 shstrLHCndMean(268,720,80)
  REAL*4 otherLHCndMean(268,720,80)
  REAL*4 allQ1RCndMean(268,720,80)
  REAL*4 convQ1RCndMean(268,720,80)
  REAL*4 dpstrQ1RCndMean(268,720,80)
  REAL*4 shstrQ1RCndMean(268,720,80)
  REAL*4 otherQ1RCndMean(268,720,80)
5.82 3HSLHT - Monthly Spectral Latent Heating

3HSLHT, "Monthly Spectral Latent Heating", produces 0.5° x 0.5° latent heating, Q1-QR, and Q2 profiles from PR rain. The PI is Dr. Takayabu and the Co-PI is Dr. Shige. The granule size is one month. The following sections describe the structure and contents of the format.

Dimension definitions:
- nlat 268 Number of 0.5° grid intervals of latitude from 67°S to 67°N.
- nlon 720 Number of 0.5° grid intervals of longitude from 180°W to 180°E.
- nlayer 80 Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1147 through Figure 1149 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputFileNames** (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

**InputAlgorithmVersions** (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

**InputGenerationDateTimes** (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.
Figure 1147: Data Format Structure for 3HSLHT, Monthly Spectral Latent Heating
### Data Format Structure for 3HSLHT, Monthly Spectral Latent Heating

<table>
<thead>
<tr>
<th>Array Name</th>
<th>Type</th>
<th>Dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1RCndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>Q1RCndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>convQ1RCndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>convQ1RCndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>dpstrQ1RCndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>dpstrQ1RCndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>shstrQ1RCndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>shstrQ1RCndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>otherQ1RCndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>otherQ1RCndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>Q2CndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>Q2CndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>convQ2CndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>convQ2CndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>dpstrQ2CndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>dpstrQ2CndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>shstrQ2CndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>shstrQ2CndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>otherQ2CndMean</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>otherQ2CndStdv</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>allPix</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>precipPix</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
<tr>
<td>convPix</td>
<td>4 bytes</td>
<td>nlat x nlon x nlayer</td>
</tr>
</tbody>
</table>

Figure 1148: Data Format Structure for 3HSLHT, Monthly Spectral Latent Heating
continued from last figure

![Diagram](image.png)

**Figure 1149: Data Format Structure for 3HSLHT, Monthly Spectral Latent Heating**

**FileInfo (Metadata):**
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**JAXAInfo (Metadata):**
JAXAInfo contains metadata requested by JAXA. Used by DPR algorithms and GSMaP. See Metadata for GPM Products for details.

**Grid (Grid)**

**GridHeader (Metadata):**
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**LHUnCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating unconditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**LHUnCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating unconditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**Q1RUnCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR unconditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**Q1RUnCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR unconditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**Q2UnCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 unconditional mean. Values range from -400 to 400 K/hr. Special values are defined
as:
-9999.9  Missing value

**Q2UnCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 unconditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**LHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**LHCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**convLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**convLHCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating convective conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**dpstrLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating deep-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**dpstrLHCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating deep-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**shstrLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**shstrLHCndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating shallow-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**otherLHCndMean** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating other conditional mean. Values range from -400 to 400 K/hr. Special
STANDARD GPM PRODUCTS

Values are defined as:
-9999.9 Missing value

otherLHCndStdv  (4-byte float, array size: nlat x nlon x nlayer):
Latent heating other conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

Q1RCndMean  (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

Q1RCndStdv  (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

convQ1RCndMean  (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

convQ1RCndStdv  (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR convective conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

dpstrQ1RCndMean  (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR deep-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

dpstrQ1RCndStdv  (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR deep-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

shstrQ1RCndMean  (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

shstrQ1RCndStdv  (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR shallow-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

otherQ1RCndMean  (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR other conditional mean. Values range from -400 to 400 K/hr. Special values are
defined as:
-9999.9 Missing value

**otherQ1CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q1 - QR other conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**Q2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**Q2CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**convQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 convective conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**convQ2CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 convective conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**dpstrQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 deep-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**dpstrQ2CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 deep-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**shstrQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 shallow-stratiform conditional mean. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**shstrQ2CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 shallow-stratiform conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**otherQ2CndMean** (4-byte float, array size: nlat x nlon x nlayer):
Q2 other conditional mean. Values range from -400 to 400 K/hr. Special values are
defined as:
-9999.9 Missing value

**otherQ2CndStdv** (4-byte float, array size: nlat x nlon x nlayer):
Q2 other conditional standard deviation. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9 Missing value

**allPix** (4-byte float, array size: nlat x nlon x nlayer):
All pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

**precipPix** (4-byte float, array size: nlat x nlon x nlayer):
The number of precipitating pixels. (= convPix + dpstrPix + shstrPix + otherPix)
Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

**convPix** (4-byte float, array size: nlat x nlon x nlayer):
Convective pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

**dpstrPix** (4-byte float, array size: nlat x nlon x nlayer):
Deep-stratiform pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

**shstrPix** (4-byte float, array size: nlat x nlon x nlayer):
Shallow-stratiform pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

**otherPix** (4-byte float, array size: nlat x nlon x nlayer):
Other pixel counts. Values range from 0 to 2000000000. Special values are defined as:
-9999.9 Missing value

C Structure Header file:

```c
#ifndef _TK_3HSLHT_H_
#define _TK_3HSLHT_H_

#ifndef _L3HSLHT_GRID_
#define _L3HSLHT_GRID_

typedef struct {
  float LHUnCndMean[80][720][268];
  float LHUnCndStdv[80][720][268];
  float Q1RUnCndMean[80][720][268];

```
float Q1UnCndStdv[80][720][268];
float Q2UnCndMean[80][720][268];
float Q2UnCndStdv[80][720][268];
float LHCndMean[80][720][268];
float LHCndStdv[80][720][268];
float convLHCndMean[80][720][268];
float convLHCndStdv[80][720][268];
float dpstrLHCndMean[80][720][268];
float dpstrLHCndStdv[80][720][268];
float shstrLHCndMean[80][720][268];
float shstrLHCndStdv[80][720][268];
float otherLHCndMean[80][720][268];
float otherLHCndStdv[80][720][268];
float Q1RcndMean[80][720][268];
float Q1RcndStdv[80][720][268];
float convQ1RcndMean[80][720][268];
float convQ1RcndStdv[80][720][268];
float dpstrQ1RcndMean[80][720][268];
float dpstrQ1RcndStdv[80][720][268];
float shstrQ1RcndMean[80][720][268];
float shstrQ1RcndStdv[80][720][268];
float otherQ1RcndMean[80][720][268];
float otherQ1RcndStdv[80][720][268];
float Q2CndMean[80][720][268];
float Q2CndStdv[80][720][268];
float convQ2CndMean[80][720][268];
float convQ2CndStdv[80][720][268];
float dpstrQ2CndMean[80][720][268];
float dpstrQ2CndStdv[80][720][268];
float shstrQ2CndMean[80][720][268];
float shstrQ2CndStdv[80][720][268];
float otherQ2CndMean[80][720][268];
float otherQ2CndStdv[80][720][268];
float allPix[80][720][268];
float precipPix[80][720][268];
float convPix[80][720][268];
float dpstrPix[80][720][268];
float shstrPix[80][720][268];
float otherPix[80][720][268];
}

L3HSLHT_GRID;

#endif
#endif

Fortran Structure Header file:

```fortran
STRUCTURE /L3HSLHT_GRID/
  REAL*4 LHUnCndMean(268,720,80)
  REAL*4 LHUnCndStdv(268,720,80)
  REAL*4 Q1RUNCndMean(268,720,80)
  REAL*4 Q1RUNCndStdv(268,720,80)
  REAL*4 Q2UNCndMean(268,720,80)
  REAL*4 Q2UNCndStdv(268,720,80)
  REAL*4 LHCndMean(268,720,80)
  REAL*4 LHCndStdv(268,720,80)
  REAL*4 convLHCndMean(268,720,80)
  REAL*4 convLHCndStdv(268,720,80)
  REAL*4 dpstrLHCndMean(268,720,80)
  REAL*4 dpstrLHCndStdv(268,720,80)
  REAL*4 shstrLHCndMean(268,720,80)
  REAL*4 shstrLHCndStdv(268,720,80)
  REAL*4 otherLHCndMean(268,720,80)
  REAL*4 otherLHCndStdv(268,720,80)
  REAL*4 Q1RCndMean(268,720,80)
  REAL*4 Q1RCndStdv(268,720,80)
  REAL*4 convQ1RCndMean(268,720,80)
  REAL*4 convQ1RCndStdv(268,720,80)
  REAL*4 dpstrQ1RCndMean(268,720,80)
  REAL*4 dpstrQ1RCndStdv(268,720,80)
  REAL*4 shstrQ1RCndMean(268,720,80)
  REAL*4 shstrQ1RCndStdv(268,720,80)
  REAL*4 otherQ1RCndMean(268,720,80)
  REAL*4 otherQ1RCndStdv(268,720,80)
  REAL*4 Q2CndMean(268,720,80)
  REAL*4 Q2CndStdv(268,720,80)
  REAL*4 convQ2CndMean(268,720,80)
  REAL*4 convQ2CndStdv(268,720,80)
  REAL*4 dpstrQ2CndMean(268,720,80)
  REAL*4 dpstrQ2CndStdv(268,720,80)
  REAL*4 shstrQ2CndMean(268,720,80)
  REAL*4 shstrQ2CndStdv(268,720,80)
  REAL*4 otherQ2CndMean(268,720,80)
  REAL*4 otherQ2CndStdv(268,720,80)
  REAL*4 allPix(268,720,80)
  REAL*4 precipPix(268,720,80)
```

REAL*4 convPix(268,720,80)
REAL*4 dpstrPix(268,720,80)
REAL*4 shstrPix(268,720,80)
REAL*4 otherPix(268,720,80)
END STRUCTURE

5.83  2HCSH - Convective Stratiform Heating

2HCSH, "Convective Stratiform Heating," produces orbital apparent heating profiles from surface convective rainfall rate and surface stratiform rainfall rate. The PI is Dr. Tao. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:
  nscan  var Number of scans in the granule.
  nray   49 Number of angle bins in each scan.
  nlayer 80 Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1150 through Figure 1151 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

NavigationRecord (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

Swath (Swath)
Figure 1150: Data Format Structure for 2HCSH, Convective Stratiform Heating

Figure 1151: Data Format Structure for 2HCSH, ScanTime
**SwathHeader** (Metadata):
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**ScanTime** (Group)
A UTC time associated with the scan.

**Year** (2-byte integer, array size: nscan):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
- 9999 Missing value

**Month** (1-byte integer, array size: nscan):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
- 99 Missing value

**DayOfMonth** (1-byte integer, array size: nscan):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
- 99 Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
- 99 Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
- 99 Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
- 99 Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
- 9999 Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
- 9999 Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. `scanTime_sec` is expressed as the UTC seconds of the day. Values range from 0 to 86400 s. Special values are defined as:
- 9999.9 Missing value

**Latitude** (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsoid. Latitude is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

**Longitude** (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsoid. Longitude is positive east, negative west. A point on the 180th meridian has the value -180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**latentHeating** (4-byte float, array size: nlayer x nray x nscan):
Latent Heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**eddyHeating** (4-byte float, array size: nlayer x nray x nscan):
Eddy flux heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**radiativeHeating** (4-byte float, array size: nlayer x nray x nscan):
Radiative heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**eddyMoistening** (4-byte float, array size: nlayer x nray x nscan):
Apparent moistening due to eddy processes. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**microMoistening** (4-byte float, array size: nlayer x nray x nscan):
Apparent moistening due to microphysical processes. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**rainType** (2-byte integer, array size: nray x nscan):
Rain type from Level 2 PR Rain Type. Special values are defined as:
-9999  Missing value

**surfacePrecipRate** (4-byte float, array size: nray x nscan):
Mean estimated surface precipitation rate from Level 2 Combined. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9  Missing value

**C Structure Header file:**

```c
#ifndef _TK_2HCSH_H_
define _TK_2HCSH_H_

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
```
short Year;
signed char Month;
signed char DayOfMonth;
signed char Hour;
signed char Minute;
signed char Second;
short MilliSecond;
short DayOfYear;
double SecondOfDay;
}

#endif

#ifndef _L2HCSH_SWATH_
#define _L2HCSH_SWATH_

typedef struct {
    SCANTIME ScanTime;
    float Latitude[49];
    float Longitude[49];
    float latentHeating[49][80];
    float eddyHeating[49][80];
    float radiativeHeating[49][80];
    float eddyMoistening[49][80];
    float microMoistening[49][80];
    short rainType[49];
    float surfacePrecipRate[49];
} L2HCSH_SWATH;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
INTEGER*2 DayOfYear
REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2HCSH_SWATH/
  RECORD /SCANTIME/ ScanTime
  REAL*4 Latitude(49)
  REAL*4 Longitude(49)
  REAL*4 latentHeating(80,49)
  REAL*4 eddyHeating(80,49)
  REAL*4 radiativeHeating(80,49)
  REAL*4 eddyMoistening(80,49)
  REAL*4 microMoistening(80,49)
  INTEGER*2 rainType(49)
  REAL*4 surfacePrecipRate(49)
END STRUCTURE

5.84 3GCSH - Gridded Orbital Convective Stratiform Heating from Combined

3GCSH, "Gridded Orbital Convective Stratiform Heating from Combined", produces 0.25° x 0.25° orbital apparent heating profiles from surface convective rainfall rate and surface stratiform rainfall rate. The PI is Dr. Wei-Kuo Tao. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:

nlat 536 Number of 0.25° grid intervals of latitude from 67°S to 67°N.
nlon 1440 Number of 0.25° grid intervals of longitude from 180°W to 180°E.
nlayer 80 Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1152 through Figure 1153 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

FileHeader (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

InputRecord (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.
Figure 1152: Data Format Structure for 3GCSH, Gridded Orbital Convective Stratiform Heating from Combined

Figure 1153: Data Format Structure for 3GCSH, GridTime
**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**Grid** (Grid)

**GridHeader** (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**latentHeating** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

**eddyHeating** (4-byte float, array size: nlat x nlon x nlayer):
Eddy flux heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

**radiativeHeating** (4-byte float, array size: nlat x nlon x nlayer):
Radiative heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

**eddyMoistening** (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to eddy processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

**microMoistening** (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to microphysical processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

**numberOfSamples** (4-byte integer, array size: nlat x nlon):
Number of samples in 0.25° x 0.25° boxes. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

**surfacePrecipRate** (4-byte float, array size: nlat x nlon):
Mean estimated surface precipitation rate from Level 2 Combined. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

**stratiformFraction** (4-byte float, array size: nlat x nlon):
Ratio of stratiform to total surface rain rate from Level 2 PR. Values range from 0 to 1.
Special values are defined as:
-9999.9  Missing value

**GridTime (Group)**
A UTC time associated with the grid box.

**Year** (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999  Missing value

**Month** (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99  Missing value

**DayOfMonth** (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

**Hour** (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

**Minute** (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

**Second** (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

**MilliSecond** (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

**DayOfYear** (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

**C Structure Header file:**

```c
#ifndef _TK_3GCSH_H_
define _TK_3GCSH_H_
#endif

#ifndef _L3GCSH_GRIDTIME_
define _L3GCSH_GRIDTIME_
#endif
```
typedef struct {
    short Year[1440][536];
    signed char Month[1440][536];
    signed char DayOfMonth[1440][536];
    signed char Hour[1440][536];
    signed char Minute[1440][536];
    signed char Second[1440][536];
    short MilliSecond[1440][536];
    short DayOfYear[1440][536];
} L3GCSH_GRIDTIME;

#endif

#ifndef _L3GCSH_GRID_
#define _L3GCSH_GRID_

typedef struct {
    float latentHeating[80][1440][536];
    float eddyHeating[80][1440][536];
    float radiativeHeating[80][1440][536];
    float eddyMoistening[80][1440][536];
    float microMoistening[80][1440][536];
    int numberOfSamples[1440][536];
    float surfacePrecipRate[1440][536];
    float stratiformFraction[1440][536];
    L3GCSH_GRIDTIME GridTime;
} L3GCSH_GRID;

#endif

#endif

Fortran Structure Header file:

STRUCTURE /L3GCSH_GRIDTIME/
    INTEGER*2 Year(536,1440)
    BYTE Month(536,1440)
    BYTE DayOfMonth(536,1440)
    BYTE Hour(536,1440)
    BYTE Minute(536,1440)
    BYTE Second(536,1440)
    INTEGER*2 MilliSecond(536,1440)
    INTEGER*2 DayOfYear(536,1440)
3HCSH, "Monthly Convective Stratiform Heating from Combined", produces 0.25° x 0.25° monthly apparent heating profiles from surface convective rainfall rate and surface stratiform rainfall rate. The PI is Dr. Wei-Kuo Tao. The granule size is one month. The following sections describe the structure and contents of the format.

Dimension definitions:
- **nlat**: 536  Number of 0.25° grid intervals of latitude from 67°S to 67°N.
- **nlon**: 1440 Number of 0.25° grid intervals of longitude from 180°W to 180°E.
- **nlayer**: 80  Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1154 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputFileNames** (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

**InputAlgorithmVersions** (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

**InputGenerationDateTimes** (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata for GPM Products for details.
Figure 1154: Data Format Structure for 3HCSH, Monthly Convective Stratiform Heating from Combined
FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

latentHeating (4-byte float, array size: nlat x nlon x nlayer):
Latent heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

eddyHeating (4-byte float, array size: nlat x nlon x nlayer):
Eddy flux heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

radiativeHeating (4-byte float, array size: nlat x nlon x nlayer):
Radiative heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

eddyMoistening (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to eddy processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

microMoistening (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to microphysical processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

numberOfSamples (4-byte integer, array size: nlat x nlon):
Number of samples in 0.25° x 0.25° boxes for one month. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

surfacePrecipRate (4-byte float, array size: nlat x nlon):
Monthly estimated surface precipitation rate from Level 3 combined. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

stratiformFraction (4-byte float, array size: nlat x nlon):
Ratio of stratiform to total surface rain rate from Level 3 PR. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

C Structure Header file:
```c
#ifndef _TK_3HCSH_H_
#define _TK_3HCSH_H_

#ifndef _L3HCSH_GRID_
#define _L3HCSH_GRID_

typedef struct {
    float latentHeating[80][1440][536];
    float eddyHeating[80][1440][536];
    float radiativeHeating[80][1440][536];
    float eddyMoistening[80][1440][536];
    float microMoistening[80][1440][536];
    int numberOfSamples[1440][536];
    float surfacePrecipRate[1440][536];
    float stratiformFraction[1440][536];
} L3HCSH_GRID;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /L3HCSH_GRID/
    REAL*4 latentHeating(536,1440,80)
    REAL*4 eddyHeating(536,1440,80)
    REAL*4 radiativeHeating(536,1440,80)
    REAL*4 eddyMoistening(536,1440,80)
    REAL*4 microMoistening(536,1440,80)
    INTEGER*4 numberOfSamples(536,1440)
    REAL*4 surfacePrecipRate(536,1440)
    REAL*4 stratiformFraction(536,1440)
END STRUCTURE
```

### 5.86 2HCSHT - Convective Stratiform Heating

2HCSHT, "Convective Stratiform Heating," produces orbital apparent heating profiles from surface convective rainfall rate and surface stratiform rainfall rate. The PI is Dr. Tao. The granule size is one orbit. The following sections describe the structure and contents of the format.
Dimension definitions:

nscan  var  Number of scans in the granule.
nray  49  Number of angle bins in each scan.
nlayer  80  Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ...
         ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1155 through Figure 1156 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputRecord (Metadata):**
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord (Metadata):**
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

FileInfo (Metadata): 
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

Swath (Swath)

SwathHeader (Metadata): 
SwathHeader contains metadata for swaths. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

ScanTime (Group) 
A UTC time associated with the scan.

Year (2-byte integer, array size: nscan): 
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999   Missing value

Month (1-byte integer, array size: nscan): 
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99   Missing value

DayOfMonth (1-byte integer, array size: nscan): 
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99  Missing value

**Hour** (1-byte integer, array size: nscan):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99  Missing value

**Minute** (1-byte integer, array size: nscan):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99  Missing value

**Second** (1-byte integer, array size: nscan):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99  Missing value

**MilliSecond** (2-byte integer, array size: nscan):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

**DayOfYear** (2-byte integer, array size: nscan):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

**SecondOfDay** (8-byte float, array size: nscan):
A time associated with the scan. scanTime_sec is expressed as the UTC seconds of the
day. Values range from 0 to 86400 s. Special values are defined as:
-9999.9  Missing value

**Latitude** (4-byte float, array size: nray x nscan):
The earth latitude of the center of the IFOV at the altitude of the earth ellipsiod. Latitude
is positive north, negative south. Values range from -90 to 90 degrees. Special values are
defined as:
-9999.9  Missing value

**Longitude** (4-byte float, array size: nray x nscan):
The earth longitude of the center of the IFOV at the altitude of the earth ellipsiod.
Longitude is positive east, negative west. A point on the 180th meridian has the value
-180 degrees. Values range from -180 to 180 degrees. Special values are defined as:
-9999.9  Missing value

**latentHeating** (4-byte float, array size: nlayer x nray x nscan):
Latent Heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**eddyHeating** (4-byte float, array size: nlayer x nray x nscan):
Eddy flux heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

**radiativeHeating** (4-byte float, array size: nlayer x nray x nscan):
Radiative heating. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value
eddyMoistening (4-byte float, array size: nlayer x nray x nscan):
Apparent moistening due to eddy processes. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

microMoistening (4-byte float, array size: nlayer x nray x nscan):
Apparent moistening due to microphysical processes. Values range from -400 to 400 K/hr. Special values are defined as:
-9999.9  Missing value

rainType (2-byte integer, array size: nray x nscan):
Rain type from Level 2 PR Rain Type. Special values are defined as:
-9999  Missing value

surfacePrecipRate (4-byte float, array size: nray x nscan):
Mean estimated surface precipitation rate from Level 2 Combined. Values range from 0 to 500 mm/hr. Special values are defined as:
-9999.9  Missing value

C Structure Header file:

```c
#ifndef _TK_2HCSHT_H_
define _TK_2HCSHT_H_

#ifndef _SCANTIME_
define _SCANTIME_

typedef struct {
  short Year;
  signed char Month;
  signed char DayOfMonth;
  signed char Hour;
  signed char Minute;
  signed char Second;
  short MilliSecond;
  short DayOfYear;
  double SecondOfDay;
} SCANTIME;

#endif

#ifndef _L2HCSHT_SWATH_
define _L2HCSHT_SWATH_

typedef struct {
  SCANTIME ScanTime;
}
```

float Latitude[49];
float Longitude[49];
float latentHeating[49][80];
float eddyHeating[49][80];
float radiativeHeating[49][80];
float eddyMoistening[49][80];
float microMoistening[49][80];
short rainType[49];
float surfacePrecipRate[49];
} L2HCSHT_SWATH;
#endif
#endif

Fortran Structure Header file:

STRUCTURE /SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
    REAL*8 SecondOfDay
END STRUCTURE

STRUCTURE /L2HCSHT_SWATH/
    RECORD /SCANTIME/ ScanTime
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    REAL*4 latentHeating(80,49)
    REAL*4 eddyHeating(80,49)
    REAL*4 radiativeHeating(80,49)
    REAL*4 eddyMoistening(80,49)
    REAL*4 microMoistening(80,49)
    INTEGER*2 rainType(49)
    REAL*4 surfacePrecipRate(49)
END STRUCTURE
5.87 3GCSHT - Gridded Orbital Convective Stratiform Heating from Combined

3GCSHT, "Gridded Orbital Convective Stratiform Heating from Combined", produces 0.25° x 0.25° orbital apparent heating profiles from surface convective rainfall rate and surface stratiform rainfall rate. The PI is Dr. Wei-Kuo Tao. The granule size is one orbit. The following sections describe the structure and contents of the format.

Dimension definitions:

- nlat 536 Number of 0.25° grid intervals of latitude from 67°S to 67°N.
- nlon 1440 Number of 0.25° grid intervals of longitude from 180°W to 180°E.
- nlayer 80 Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1157 through Figure 1158 show the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader (Metadata):**
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.
**InputRecord** (Metadata):
InputRecord contains a record of input files for this granule. This group appears in Level 1 and Level 2 data products. Level 3 time averaged products have the same information separated into 3 groups since they have many inputs. See Metadata for GPM Products for details.

**NavigationRecord** (Metadata):
NavigationRecord contains navigation metadata for this granule. This group appears in Level 1 and Level 2 data products. See Metadata for GPM Products for details.

**FileInfo** (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

**Grid** (Grid)

**GridHeader** (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

**latentHeating** (4-byte float, array size: nlat x nlon x nlayer):
Latent heating. Values range from -50 to 100 K/hr. Special values are defined as:
- 9999.9 Missing value

**eddyHeating** (4-byte float, array size: nlat x nlon x nlayer):
Eddy flux heating. Values range from -50 to 100 K/hr. Special values are defined as:
- 9999.9 Missing value

**radiativeHeating** (4-byte float, array size: nlat x nlon x nlayer):
Radiative heating. Values range from -50 to 100 K/hr. Special values are defined as:
- 9999.9 Missing value
**eddyMoistening** (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to eddy processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

**microMoistening** (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to microphysical processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

**numberOfSamples** (4-byte integer, array size: nlat x nlon):
Number of samples in 0.25° x 0.25° boxes. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

**surfacePrecipRate** (4-byte float, array size: nlat x nlon):
Mean estimated surface precipitation rate from Level 2 Combined. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

**stratiformFraction** (4-byte float, array size: nlat x nlon):
Ratio of stratiform to total surface rain rate from Level 2 PR. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value

---

**GridTime** (Group)
A UTC time associated with the grid box.

**Year** (2-byte integer, array size: nlat x nlon):
4-digit year, e.g., 1998. Values range from 1950 to 2100 years. Special values are defined as:
-9999 Missing value

**Month** (1-byte integer, array size: nlat x nlon):
Month of the year. Values range from 1 to 12 months. Special values are defined as:
-99 Missing value

**DayOfMonth** (1-byte integer, array size: nlat x nlon):
Day of the month. Values range from 1 to 31 days. Special values are defined as:
-99 Missing value

**Hour** (1-byte integer, array size: nlat x nlon):
UTC hour of the day. Values range from 0 to 23 hours. Special values are defined as:
-99 Missing value

**Minute** (1-byte integer, array size: nlat x nlon):
Minute of the hour. Values range from 0 to 59 minutes. Special values are defined as:
-99 Missing value
Second (1-byte integer, array size: nlat x nlon):
Second of the minute. Values range from 0 to 60 s. Special values are defined as:
-99    Missing value

MilliSecond (2-byte integer, array size: nlat x nlon):
Thousandths of the second. Values range from 0 to 999 ms. Special values are defined as:
-9999  Missing value

DayOfYear (2-byte integer, array size: nlat x nlon):
Day of the year. Values range from 1 to 366 days. Special values are defined as:
-9999  Missing value

C Structure Header file:

```c
#ifndef _TK_3GCSHT_H_
#define _TK_3GCSHT_H_

#ifndef _L3GCSHT_GRIDTIME_
#define _L3GCSHT_GRIDTIME_

typedef struct {
    short Year[1440][536];
    signed char Month[1440][536];
    signed char DayOfMonth[1440][536];
    signed char Hour[1440][536];
    signed char Minute[1440][536];
    signed char Second[1440][536];
    short MilliSecond[1440][536];
    short DayOfYear[1440][536];
} L3GCSHT_GRIDTIME;

#endif

#ifndef _L3GCSHT_GRID_
#define _L3GCSHT_GRID_

typedef struct {
    float latentHeating[80][1440][536];
    float eddyHeating[80][1440][536];
    float radiativeHeating[80][1440][536];
    float eddyMoistening[80][1440][536];
    float microMoistening[80][1440][536];
    int numberOfSamples[1440][536];
    float surfacePrecipRate[1440][536];
    float stratiformFraction[1440][536];
}
#endif
```
L3GCSHT_GRIDTIME GridTime;
} L3GCSHT_GRID;

#endif
#endif

Fortran Structure Header file:

\begin{verbatim}
STRUCTURE /L3GCSHT_GRIDTIME/
  INTEGER*2 Year(536,1440)
  BYTE Month(536,1440)
  BYTE DayOfMonth(536,1440)
  BYTE Hour(536,1440)
  BYTE Minute(536,1440)
  BYTE Second(536,1440)
  INTEGER*2 MilliSecond(536,1440)
  INTEGER*2 DayOfYear(536,1440)
END STRUCTURE

STRUCTURE /L3GCSHT_GRID/
  REAL*4 latentHeating(536,1440,80)
  REAL*4 eddyHeating(536,1440,80)
  REAL*4 radiativeHeating(536,1440,80)
  REAL*4 eddyMoistening(536,1440,80)
  REAL*4 microMoistening(536,1440,80)
  INTEGER*4 numberOfSamples(536,1440)
  REAL*4 surfacePrecipRate(536,1440)
  REAL*4 stratiformFraction(536,1440)
END STRUCTURE
\end{verbatim}

5.88 3HCSHT - Monthly Convective Stratiform Heating from Combined

3HCSHT, "Monthly Convective Stratiform Heating from Combined", produces 0.25° x 0.25° monthly apparent heating profiles from surface convective rainfall rate and surface stratiform rainfall rate. The PI is Dr. Wei-Kuo Tao. The granule size is one month. The following sections describe the structure and contents of the format.

Dimension definitions:
5.88  3HCSHT - Monthly Convective Stratiform Heating from Combined

Figure 1159: Data Format Structure for 3HCSHT, Monthly Convective Stratiform Heating from Combined

![Diagram](image)

**Figure 1159**: Data Format Structure for 3HCSHT, Monthly Convective Stratiform Heating from Combined

- **nlat**: 536 Number of 0.25° grid intervals of latitude from 67°S to 67°N.
- **nlon**: 1440 Number of 0.25° grid intervals of longitude from 180°W to 180°E.
- **nlayer**: 80 Number of layers at the fixed heights of 0.00-0.25 km, 0.25-0.50 km, ..., 19.50-19.75 km, and 19.75-20.00 km.

Figure 1159 shows the structure of this product. The text below describes the contents of objects in the structure, the C Structure Header File and the Fortran Structure Header File.

**FileHeader** (Metadata):
FileHeader contains general metadata. This group appears in all data products. See Metadata for GPM Products for details.

**InputFileNames** (Metadata):
InputFileNames contains a list of input file names for this granule. See Metadata for GPM Products for details.

**InputAlgorithmVersions** (Metadata):
InputAlgorithmVersions contains a list of input algorithm versions for this granule. See Metadata for GPM Products for details.

**InputGenerationDateTimes** (Metadata):
InputGenerationDateTimes contains a list of input generation datetimes. See Metadata
for GPM Products for details.

FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for GPM Products for details.

Grid (Grid)

GridHeader (Metadata):
GridHeader contains metadata defining the grids in the grid structure. See Metadata for GPM Products for details.

latentHeating (4-byte float, array size: nlat x nlon x nlayer):
Latent heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

eddyHeating (4-byte float, array size: nlat x nlon x nlayer):
Eddy flux heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

radiativeHeating (4-byte float, array size: nlat x nlon x nlayer):
Radiative heating. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

eddyMoistening (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to eddy processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

microMoistening (4-byte float, array size: nlat x nlon x nlayer):
Apparent moistening due to microphysical processes. Values range from -50 to 100 K/hr. Special values are defined as:
-9999.9 Missing value

numberOfSamples (4-byte integer, array size: nlat x nlon):
Number of samples in 0.25° x 0.25° boxes for one month. Values range from 0 to 500000. Special values are defined as:
-9999 Missing value

surfacePrecipRate (4-byte float, array size: nlat x nlon):
Monthly estimated surface precipitation rate from Level 3 combined. Values range from 0 to 3000 mm/hr. Special values are defined as:
-9999.9 Missing value

stratiformFraction (4-byte float, array size: nlat x nlon):
Ratio of stratiform to total surface rain rate from Level 3 PR. Values range from 0 to 1. Special values are defined as:
-9999.9 Missing value
C Structure Header file:

```c
#ifndef _TK_3HCSHT_H_
#define _TK_3HCSHT_H_

#ifndef _L3HCSHT_GRID_
#define _L3HCSHT_GRID_

typedef struct {
    float latentHeating[80][1440][536];
    float eddyHeating[80][1440][536];
    float radiativeHeating[80][1440][536];
    float eddyMoistening[80][1440][536];
    float microMoistening[80][1440][536];
    int numberOfSamples[1440][536];
    float surfacePrecipRate[1440][536];
    float stratiformFraction[1440][536];
} L3HCSHT_GRID;

#endif
#endif

```

Fortran Structure Header file:

```
STRUCTURE /L3HCSHT_GRID/
    REAL*4 latentHeating(536,1440,80)
    REAL*4 eddyHeating(536,1440,80)
    REAL*4 radiativeHeating(536,1440,80)
    REAL*4 eddyMoistening(536,1440,80)
    REAL*4 microMoistening(536,1440,80)
    INTEGER*4 numberOfSamples(536,1440)
    REAL*4 surfacePrecipRate(536,1440)
    REAL*4 stratiformFraction(536,1440)
END STRUCTURE
```