TROPICAL RAINFALL MEASURING MISSION PRECIPITATION PROCESSING SYSTEM

File Specification
2A23

Version 7

February 20, 2014

0.1 2A23 - PR Qualitative

2A23, "PR Qualitative", produces a Rain/No-rain flag. If rain is present, this algorithm will detect the bright band, determine the heights of the bright band and the storm, and classify rain types. 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.

Figure 1 through Figure 5 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 TRMM 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 TRMM 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 TRMM 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 TRMM Products for details.

JAXAInfo (Metadata):
JAXAInfo contains metadata requested by JAXA. Used by PR algorithms only. See Metadata for TRMM 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 TRMM Products for details.

ScanTime (Group)

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

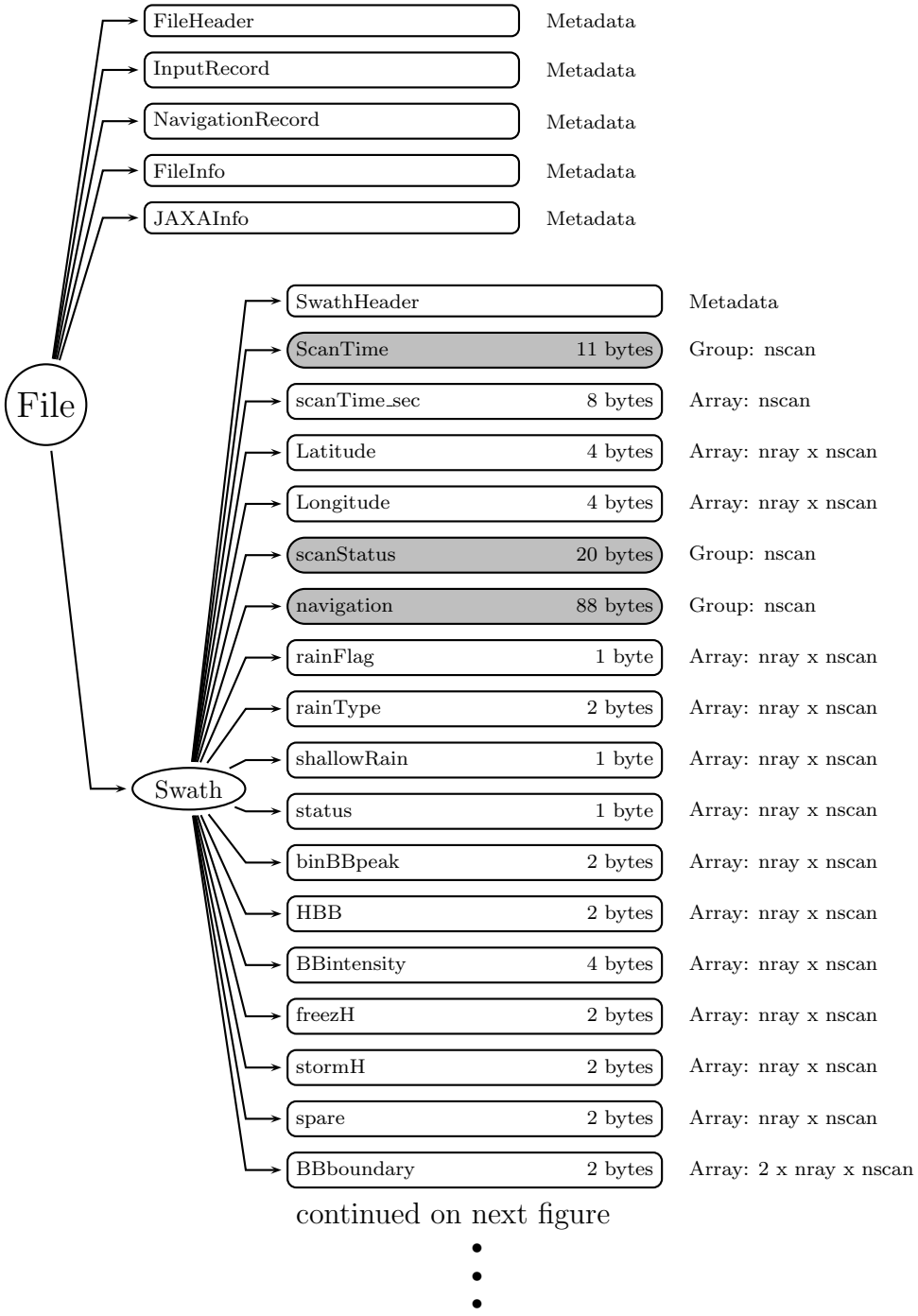

Figure 1: Data Format Structure for 2A23, PR Qualitative

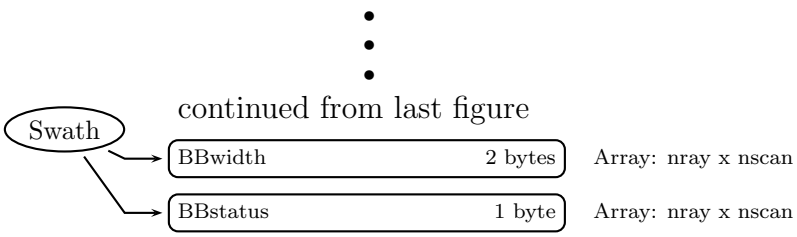

Figure 2: Data Format Structure for 2A23, PR Qualitative

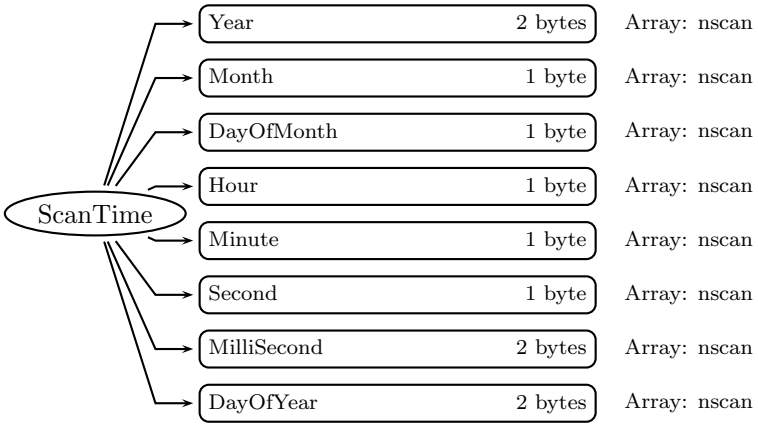

Figure 3: Data Format Structure for 2A23, ScanTime

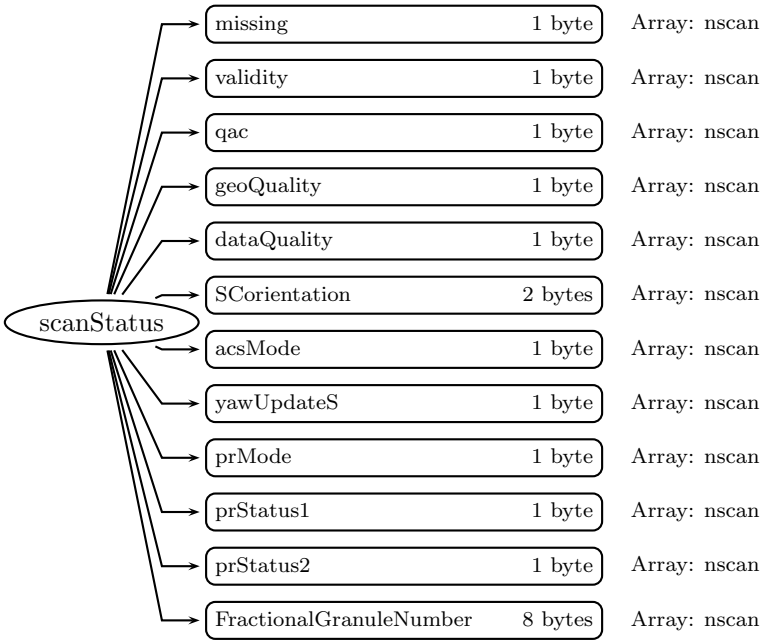

Figure 4: Data Format Structure for 2A23, scanStatus

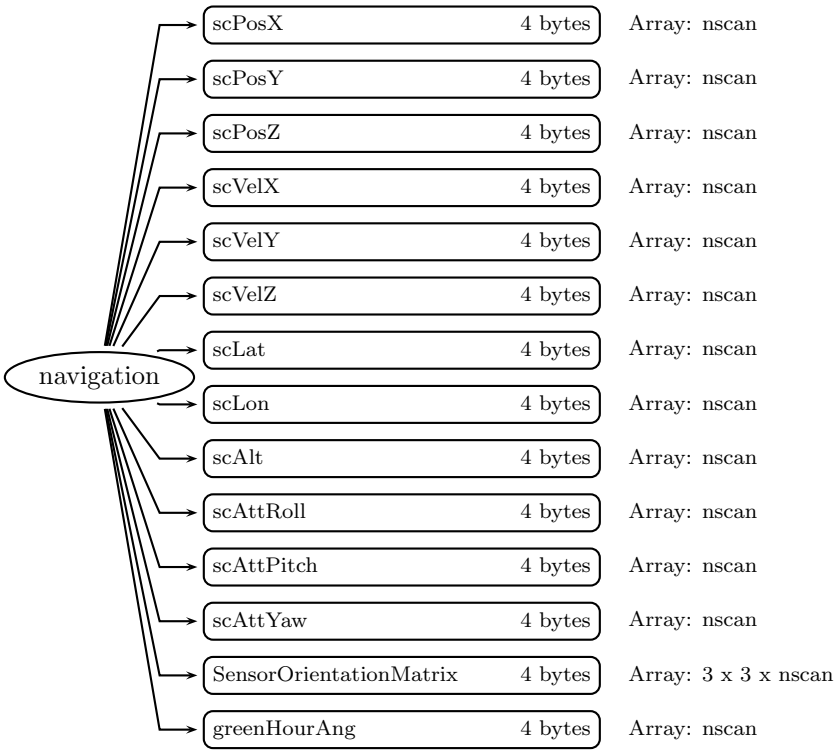

Figure 5: Data Format Structure for 2A23, navigation

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

scanTime_sec (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)

missing (1-byte integer, array size: nscan):
Missing indicates whether information is contained in the scan data. The values are:

```
0 Scan data elements contain information
1 Scan was missing in the telemetry data
2 Scan data contains no elements with rain
```

validity (1-byte integer, array size: nscan):
Validity is a summary of status modes. If all status modes are routine, all bits in Validity = 0. Routine means that scan data has been measured in the normal operational situation as far as the status modes are concerned. Validity does not assess data or geolocation quality. Validity 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   Non-routine spacecraft orientation (2 or 3)
2   Non-routine ACS mode (other than 4)
3   Non-routine yaw update status (0 or 1)
4   Non-routine instrument status (other than 1)
5   Non-routine QAC (non-zero)
6   Spare (always 0)
7   Spare (always 0)
```

qac (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.

geoQuality (1-byte integer, array size: nscan):
Geolocation quality is a summary of geolocation quality in the scan. A zero integer value indicates 'good' 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   latitude limit error
1   geolocation
2   attitude change rate limit error
3   attitude limit error
4   satellite undergoing maneuvers
5   using predictive orbit data
6   geolocation calculation error
7   not used
```

dataQuality (1-byte integer, array size: nscan):
Data quality is a summary of data quality in the scan. Unless this is 0 (normal), the scan data is meaningless to higher 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   Geolocation Quality is not normal
6   Validity is not normal
```

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 TMI scan. If +X is forward, SCorientation is 0. If -X is forward, SCorientation is 180. If -Y is forward, SCorientation is 90. Values range from 0 to 360 degrees. Special values are defined as:

-8003 Inertial
-8004 Unknown
-9999 Missing value

acsMode (1-byte integer, array size: nscan):

```
Value Meaning
```

0	Standby
1	Sun Acquire
2	Earth Acquire
3	Yaw Acquire
4	Nominal
5	Yaw Maneuver
6	Delta-H (Thruster)
7	Delta-V (Thruster)
8	CERES Calibration

yawUpdateS (1-byte integer, array size: nscan):

Value	Meaning
0	Inaccurate
1	Indeterminate
2	Accurate

prMode (1-byte integer, array size: nscan):

Value	Meaning
1	Observation Mode
2	Other Mode

prStatus1 (1-byte integer, array size: nscan):
This status is a warning for scan data. Unless this is 0, the scan data may include a little questionable value though it is not a problem (such as break of caution limit). This field is used only for NASDA's data analysis.

prStatus2 (1-byte integer, array size: nscan):
Initialization in Onboard Surface Search Algorithm.

Value	Meaning
0	Not initialized
1	Initialized

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)

scPosX (4-byte float, array size: nscan):
The x component of the position (m) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time (i.e., time at the middle pixel/IFOV of the active scan period). Geocentric Inertial Coordinates are also commonly known as Earth Centered Inertial coordinates. These coordinates will be True of Date (rather than Epoch 2000 which are also commonly used), as interpolated from the data in the Flight Dynamics Facility ephemeris files generated for TRMM.

scPosY (4-byte float, array size: nscan):
The y component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scPosZ (4-byte float, array size: nscan):
The z component of the position (m) of the spacecraft in Geocentric Inertial Coordinates. See scPosX.

scVelX (4-byte float, array size: nscan):
The x component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelY (4-byte float, array size: nscan):
The y component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scVelZ (4-byte float, array size: nscan):
The z component of the velocity (ms^{-1}) of the spacecraft in Geocentric Inertial Coordinates at the Scan mid-Time.

scLat (4-byte float, array size: nscan):
The geodedic latitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scLon (4-byte float, array size: nscan):
The geodedic longitude (decimal degrees) of the spacecraft at the Scan mid-Time.

scAlt (4-byte float, array size: nscan):
The altitude (m) of the spacecraft above the Earth Ellipsiod at the Scan mid-Time.

scAttRoll (4-byte float, array size: nscan):
The 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.

scAttPitch (4-byte float, array size: nscan):
The satellite attitude Euler pitch 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.

scAttYaw (4-byte float, array size: nscan):
The satellite attitude Euler yaw 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.

SensorOrientationMatrix (4-byte float, array size: 3 x 3 x nscan):
SensorOrientationMatrix is the rotation matrix from the instrument coordinate frame to Geocentric Inertial Coordinates at the Scan mid-Time. It is unitless.

greenHourAng (4-byte float, array size: nscan):
The rotation angle (degrees) from Geocentric Inertial Coordinates to Earth Fixed Coordinates.

rainFlag (1-byte integer, array size: nray x nscan):
The Rain Flag is identical to the Minimum Echo Flag of 1C21:

```
 0: no rain
10: rain possible
11: rain possible (echo greater than rain threshold 1 in clutter region)
12: rain possible (echo greater than rain threshold 2 in clutter region)
13: rain possible
15: rain probable
20: rain certain
```

rainType (2-byte integer, array size: nray x nscan):
The Rain Type is set as follows:

```
100: Stratiform.
When R\_type\_V[i] = T\_stra,
and R\_type\_H[i] = T\_stra.
(BB detected.)

105: Stratiform.
--- added in V7.
When R\_type\_V[i] = T\_stra,
(BB detected.)
and R\_type\_H[i] = T\_stra,
But storm top (determined by 2A23) is too high.

110: Stratiform.
When R\_type\_V[i] = T\_stra,
(BB detected.)
and R\_type\_H[i] = T\_other.

115: Stratiform.
--- added in V7.
When R\_type\_V[i] = T\_stra,
(BB detected.)
and R\_type\_H[i] = T\_other.
But storm top (determined by 2A23) is too high.

120: Probably stratiform. (BB may exist but not detected.)
When R\_type\_V[i] = T\_other,
and R\_type\_H[i] = T\_stra.

130: Maybe stratiform.
When R\_type\_V[i] = T\_stra,
and R\_type\_H[i] = T\_conv.
(BB detected.)

135: Maybe stratiform.
--- added in V7.
When R\_type\_V[i] = T\_stra,
(BB detection certain.)
and R\_type\_H[i] = T\_conv.
But storm top (determined by 2A23) is too high.

140: Maybe stratiform. (BB hardly expected.)
```

When R_type_V[i] = T_other,
and R_type_H[i] = T_stra.

152: Maybe stratiform:
When R_type_V[i] = T_other,
R_type_H[i] = T_stra,
and shallowRain[i] = 20 or 21.
(Shallow non-isolated is detected.)

160: Maybe stratiform, but rain hardly expected near surface.
BB may exist but is not detected.
When R_type_V[i] = T_other,
and R_type_H[i] = T_stra.

170: Maybe stratiform, but rain hardly expected near surface.
BB hardly expected. Maybe cloud only.
Distinction between 170 and 300 is very small.
When R_type_V[i] = T_other,
and R_type_H[i] = T_stra.

200: Convective.
When R_type_V[i] = T_conv,
and R_type_H[i] = T_conv.
210: Convective.
When R_type_V[i] = T_other,
and R_type_H[i] = T_conv;

220: Convective
When R_type_V[i] = T_conv,
and R_type_H[i] = T_other;

230: Probably convective. ---> Re-introduced in V7.
When R_type_V[i] = T_stra;
(BB exists)
R_type_H[i] = T_conv;
and Z below BB is strong.

235: Probably convective. ---> Added in V7.
When R_type_V[i] = T_other;
R_type_H[i] = T_stra;
But storm top (determined by 2A23) is too high.

237: Probably convective. ---> Added in V7.

When R_type_V[i] = T_other;
R_type_H[i] = T_stra;
But the cell size is small.

240: Maybe convective.
When R_type_V[i] = T_conv,
and R_type_H[i] = T_stra;

251: Convective.
When R_type_V[i] = T_conv,
R_type_H[i] = T_conv,
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

252: Convective.
When R_type_V[i] = T_conv,
R_type_H[i] = T_conv,
and shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)

261: Convective.
When R_type_V[i] = T_conv,
R_type_H[i] = T_conv;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

262: Convective.
When R_type_V[i] = T_conv,
R_type_H[i] = T_other;
and shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)

271: Convective.
When R_type_V[i] = T_other,
R_type_H[i] = T_conv;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

272: Convective.
When R_type_V[i] = T_other,
R_type_H[i] = T_conv;
and shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)

281: Convective.
When R_type_V[i] = T_conv,
R_type_H[i] = T_stra;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

282: Convective.
When R_type_V[i] = T_conv,
R_type_H[i] = T_stra;
and shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)

291: Convective:
When R_type_V[i] = T_other;
R_type_H[i] = T_stra;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

292: Convective:
--- added in V7.
When R_type_V[i] = T_other;
R_type_H[i] = T_stra;
and shallowRain[i] = 20 or 21;
Though this is shallow non-isolated, the appearance is
'sporadic', hence convective.

297: Convective:
--- added in V7.
When R_type_V[i] = T_other;
R_type_H[i] = T_stra;
shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)
But the cell size is small.

300: Other.
When R_type_V[i] = T_other;
and R_type_H[i] = T_other;
This category includes very weak echo (possibly noise)
and/or cloud.

311: Others.
--- added in V7.

```
When R\_type\_V[i] = T\_other,
R\_type\_H[i] = T\_other;
and shallowRain[i] = 10 or 11;
(Shallow isolated is detected)

312: Other.
When R\_type\_V[i] = T\_other,
R\_type\_H[i] = T\_other;
and shallowRain[i] = 20 or 21;
(Shallow non-isolated is detected)

313: Other.
When R\_type\_V[i] = T\_other,
R\_type\_H[i] = T\_other;
If sidelobe clutter were not rejected.

-88: no rain

-99: missing
```

The above assignment of numbers has the following meaning:

shallowRain (1-byte integer, array size: nray x nscan):
The Shallow Rain Flag takes the following values:

```
shallowRain(i)  = 10: maybe shallow, isolated,
                = 11: shallow isolated (with confidence),
                = 20: maybe shallow but not isolated,
                = 21: shallow but not isolated (width confidence)
                =  0: when not shallow.
                less than  0: when not rain certain or data missing.
```

status (1-byte integer, array size: nray x nscan):
The Status Flag indicates whether the data are obtained over sea or land and the confidence of 2A-23 product data. It is set as follows:

```
 0: good (over ocean)
10: BB detection may be good (over ocean)
20: R-type classification may be good (over ocean)
    (BB detection is good or BB does not exist)
30: Both BB detection and R-type classification may be
    good (over ocean)
50: not good (because of warnings)  (over ocean)
```

```
100: bad (possible data corruption)  (over ocean)

  1: good (over land)
 11: BB detection may be good (over land)
 21: R-type classification may be good (over land)
     (BB detection is good or BB does not exist)
 31: Both BB detection and R-type classification may be
     good (over land)
 51: not good (because of warnings)  (over land)
101: bad (possible data corruption)  (over land)

  2: good (over coastline)
 12: BB detection may be good (over coastline)
 22: R-type classification may be good (over coastline)
     (BB detection is good or BB does not exist)
 32: Both BB detection and R-type classification may be
     good (over coastline)
 52: not good (because of warnings)  (over coastline)
102: bad (possible data corruption)  (over coastline)

  4: good (over inland lake)
 14: BB detection may be good (over inland lake)
 24: R-type classification may be good (over inland lake)
     (BB detection is good or BB does not exist)
 34: Both BB detection and R-type classification may be
     good (over inland lake)
 54: not good (because of warnings)  (over inland lake)
104: bad (possible data corruption)  (over inland lake)

  9: may be good (land/sea unknown)
 19: BB detection may be good  (land/sea unknown)
 29: R-type classification may be good (BB detection is
     good or BB does not exist)      (land/sea unknown)
 39: Both BB detection and R-type classification may be
     good (land/sea unknown)
 59: not good (because of warnings)  (land/sea unknown)
109: bad (possible data corruption)  (land/sea unknown)

When it is "no rain" or "data missing",
Status Flag contains the following values:
   -88:    no rain
   -99:    data missing
```

```
Assignment of the above numbers are based on the following rules:
   When Status
   Status/100   =   0:  good, may be good, or not good
                    1:  doubtful
```

binBBpeak (2-byte integer, array size: nray x nscan):
A positive range bin number that corresponds to the peak of the bright band. This bin number is in the Level-1 bin numbering scheme (125m, see Level-1 PR description). Values range from 1 to 400. Special values are defined as:
-8888 No rain
-1111 No bright band
-9999 Missing value

HBB (2-byte integer, array size: nray x nscan):
A positive Height of Bright Band is defined in meters above mean sea level. Values are in m. Special values are defined as:
-8888 No rain
-1111 No bright band
-9999 Missing value

BBintensity (4-byte float, array size: nray x nscan):
The maximum value of the bright band obtained from normal samples. Values range from 0.00 to 100.0 dBZ. Special values are defined as:
-8888 No rain
-1111 No bright band
-9999 Missing value

freezH (2-byte integer, array size: nray x nscan):
A positive Height of Freezing Level is the height of the 0^{o}C isotherm above mean sea level, estimated from GANAL (Global analysis data by Japanese Meteorological Agency) surface temperature data. Values are in m. Special values are defined as:
-8888 No rain
-5555 When error occurred in the estimation of Height of Freezing Level
-9999 Missing value

stormH (2-byte integer, array size: nray x nscan):
A positive Height of Storm is the height of the storm top above mean sea level. A positive Height of Storm is given only when rain is present with a high degree of confidence in 1C21 (i.e., the Minimum Echo Flag in 1C21 has the value of 2 (rain certain)). Values range from 0 to 30000 m. Special values are defined as:
-8888 No rain
-1111 Rain is not present with a high level of confidence in 1C21
-9999 Missing value

spare (2-byte integer, array size: nray x nscan):
Contains developer output.

BBboundary (2-byte integer, array size: 2 x nray x nscan):
Positive bin numbers of the boundary of the bright band. The first index indicates the top of the bright band, the second index indicates the bottom. These bin numbers are in the Level-1 bin numbering scheme (125m, see Level-1 PR description). Values range from 0.00 to 100.0. Special values are defined as:
-8888 No rain
-1111 No bright band
-9999 Missing value

BBwidth (2-byte integer, array size: nray x nscan):
Width of the bright band. Values are in m. Special values are defined as:
-8888 No rain
-1111 No bright band
-9999 Missing value

BBstatus (1-byte integer, array size: nray x nscan):
Indicates the status of the bright band detection. This flag is a composite of three internal status flags:

```
BB\_status(j) = BB\_detection\_status(j) * 16
              + BB\_boundary\_status(j) * 4
              + BB\_width\_status(j)

where each status on the right hand side takes the following
values:
      1: poor,
      2: fair,
      3: good.


These three internal flags would be computed from BB\_status(j),
for example, by something like as follows:

if (BB\_status(j)>0)
{
    BB\_detection\_status(j) = BB\_status(j) / 16;
    BB\_boundary\_status(j) = (BB\_status(j)%16) / 4;
    BB\_width\_status(j) = BB\_status(j)%4;
}

where % means MOD in FORTRAN;
```

C Structure Header file:

```
#ifndef _TK_2A23_H_
#define _TK_2A23_H_

#ifndef _L2A23_NAVIGATION_
#define _L2A23_NAVIGATION_

typedef struct {
    float scPosX;
    float scPosY;
    float scPosZ;
    float scVelX;
    float scVelY;
    float scVelZ;
    float scLat;
    float scLon;
    float scAlt;
    float scAttRoll;
    float scAttPitch;
    float scAttYaw;
    float SensorOrientationMatrix[3][3];
    float greenHourAng;
} L2A23_NAVIGATION;

#endif

#ifndef _L2A23_SCANSTATUS_
#define _L2A23_SCANSTATUS_

typedef struct {
    signed char missing;
    signed char validity;
    signed char qac;
    signed char geoQuality;
    signed char dataQuality;
    short SCorientation;
    signed char acsMode;
    signed char yawUpdateS;
    signed char prMode;
    signed char prStatus1;
    signed char prStatus2;
    double FractionalGranuleNumber;
} L2A23_SCANSTATUS;
```

```
#endif

#ifndef _L2A23_SCANTIME_
#define _L2A23_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2A23_SCANTIME;

#endif

#ifndef _L2A23_SWATH_
#define _L2A23_SWATH_

typedef struct {
    L2A23_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    L2A23_SCANSTATUS scanStatus;
    L2A23_NAVIGATION navigation;
    signed char rainFlag[49];
    short rainType[49];
    signed char shallowRain[49];
    signed char status[49];
    short binBBpeak[49];
    short HBB[49];
    float BBintensity[49];
    short freezH[49];
    short stormH[49];
    short spare[49];
    short BBboundary[49][2];
    short BBwidth[49];
    signed char BBstatus[49];
} L2A23_SWATH;
```

```
#endif

#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2A23_NAVIGATION/
    REAL*4 scPosX
    REAL*4 scPosY
    REAL*4 scPosZ
    REAL*4 scVelX
    REAL*4 scVelY
    REAL*4 scVelZ
    REAL*4 scLat
    REAL*4 scLon
    REAL*4 scAlt
    REAL*4 scAttRoll
    REAL*4 scAttPitch
    REAL*4 scAttYaw
    REAL*4 SensorOrientationMatrix(3,3)
    REAL*4 greenHourAng
END STRUCTURE

STRUCTURE /L2A23_SCANSTATUS/
    BYTE missing
    BYTE validity
    BYTE qac
    BYTE geoQuality
    BYTE dataQuality
    INTEGER*2 SCorientation
    BYTE acsMode
    BYTE yawUpdateS
    BYTE prMode
    BYTE prStatus1
    BYTE prStatus2
    REAL*8 FractionalGranuleNumber
END STRUCTURE

STRUCTURE /L2A23_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
```

```
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L2A23_SWATH/
    RECORD /L2A23_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L2A23_SCANSTATUS/ scanStatus
    RECORD /L2A23_NAVIGATION/ navigation
    BYTE rainFlag(49)
    INTEGER*2 rainType(49)
    BYTE shallowRain(49)
    BYTE status(49)
    INTEGER*2 binBBpeak(49)
    INTEGER*2 HBB(49)
    REAL*4 BBintensity(49)
    INTEGER*2 freezH(49)
    INTEGER*2 stormH(49)
    INTEGER*2 spare(49)
    INTEGER*2 BBboundary(2,49)
    INTEGER*2 BBwidth(49)
    BYTE BBstatus(49)
END STRUCTURE
```