TROPICAL RAINFALL MEASURING MISSION PRECIPITATION PROCESSING SYSTEM

File Specification 2A21

Version 7

March 23, 2012

0.1 2A21 - Surface Cross Section

2A21, "Surface Cross Section," computes the normalized surface cross section. If rain is present, it will also compute path attenuation and its associated reliability factor. Figure 1.2.1-1 shows the structure of the 2A21 product in terms of the component objects and their sizes.

Dimension definitions:

nscan	var	Number of scans in the granule.
nray	49	Number of angle bins in each scan.
refmethod	5	Number of reference methods.
direction	2	Number of refScanID directions.
distance	2	Number of refScanID distances.

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.

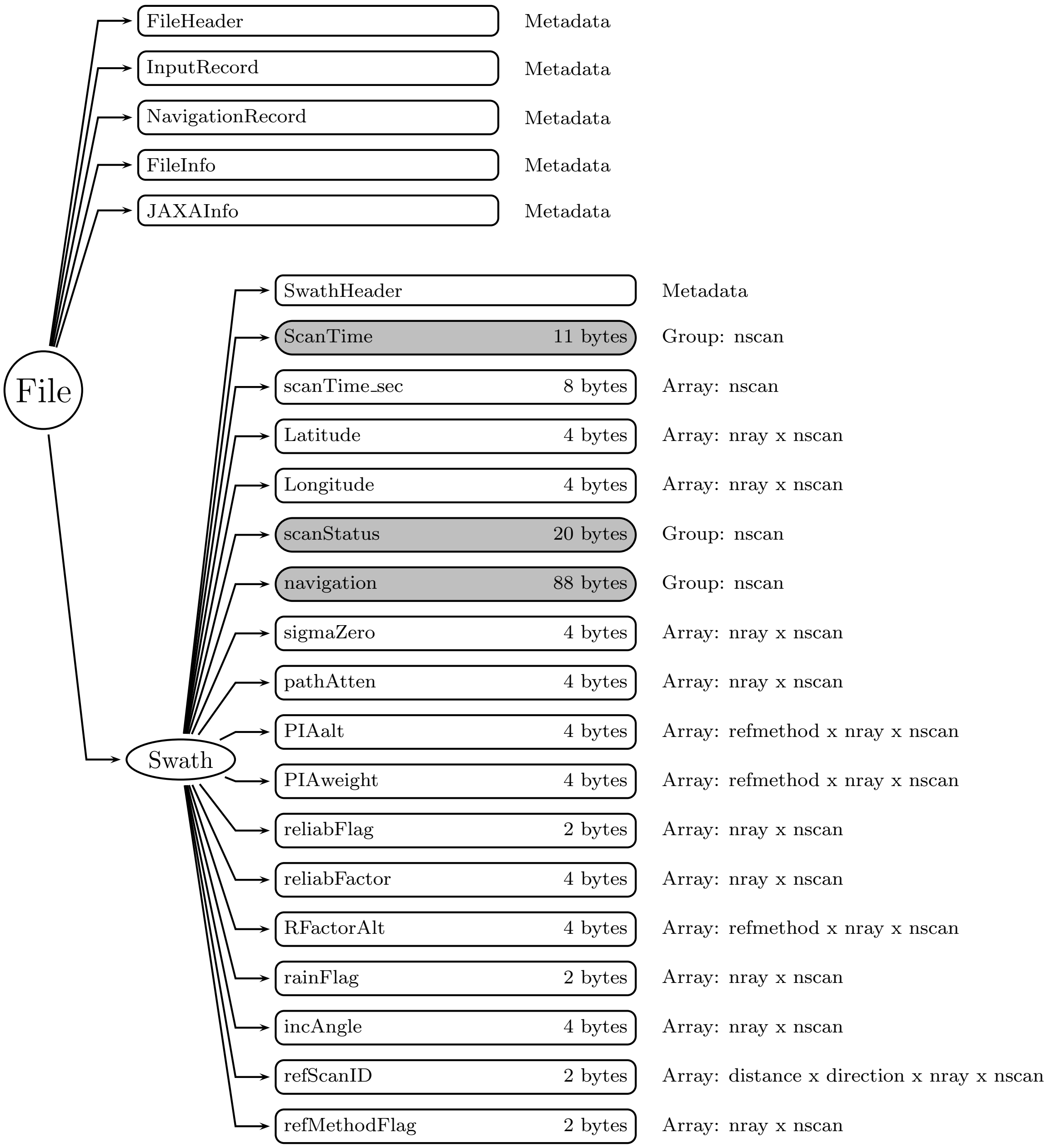
FileInfo (Metadata):
FileInfo contains metadata used by the PPS I/O Toolkit (TKIO). This group appears in all data products. See Metadata for TRMM Products for details.

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.

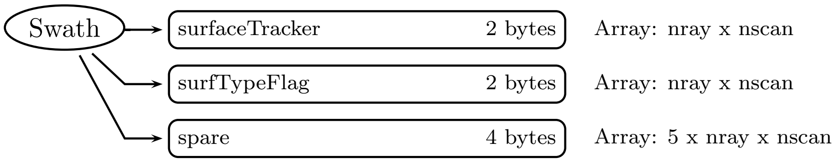
ScanTime (Group)



continued on next figure

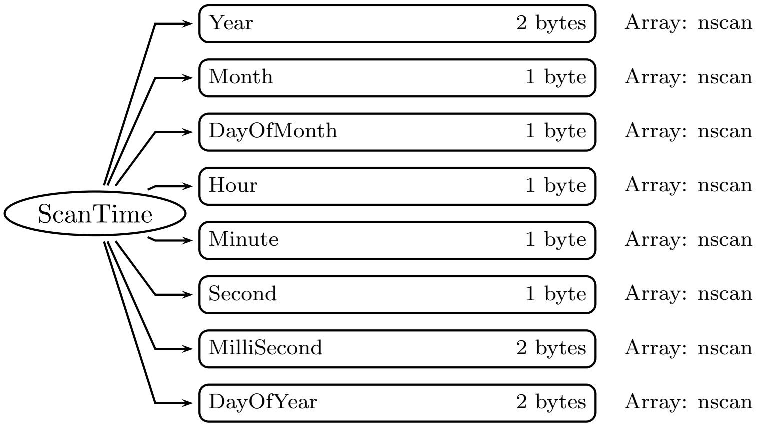
Figure 1: Data Format Structure for 2A21, Surface Cross Section

continued from last figure



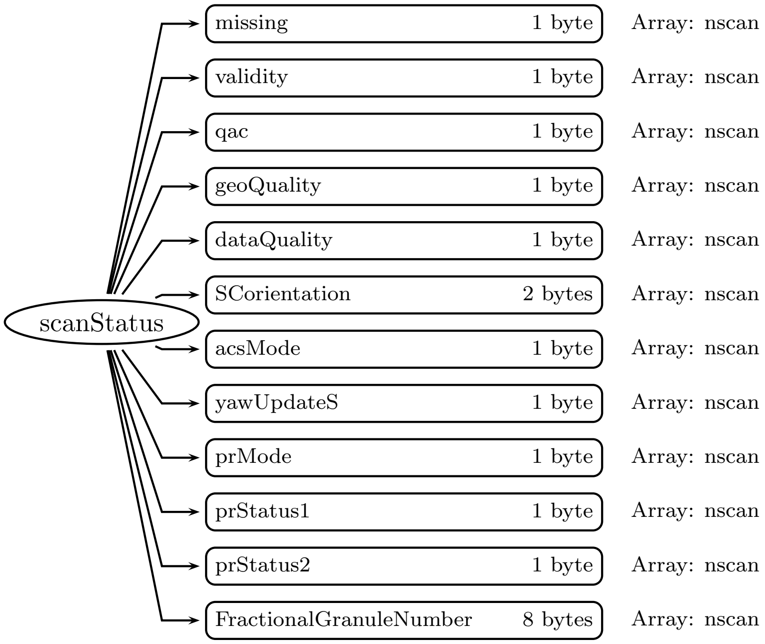
surfaceTracker	2 bytes	Array: nray x nscan
surfTypeFlag	2 bytes	Array: nray x nscan
spare	4 bytes	Array: 5 x nray x nscan

Figure 2: Data Format Structure for 2A21, Surface Cross Section



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

Figure 3: Data Format Structure for 2A21, ScanTime



missing	1 byte	Array: nscan
validity	1 byte	Array: nscan
qac	1 byte	Array: nscan
geoQuality	1 byte	Array: nscan
dataQuality	1 byte	Array: nscan
SCorientation	2 bytes	Array: nscan
acsMode	1 byte	Array: nscan
yawUpdateS	1 byte	Array: nscan
prMode	1 byte	Array: nscan
prStatus1	1 byte	Array: nscan
prStatus2	1 byte	Array: nscan
FractionalGranuleNumber	8 bytes	Array: nscan

Figure 4: Data Format Structure for 2A21, scanStatus

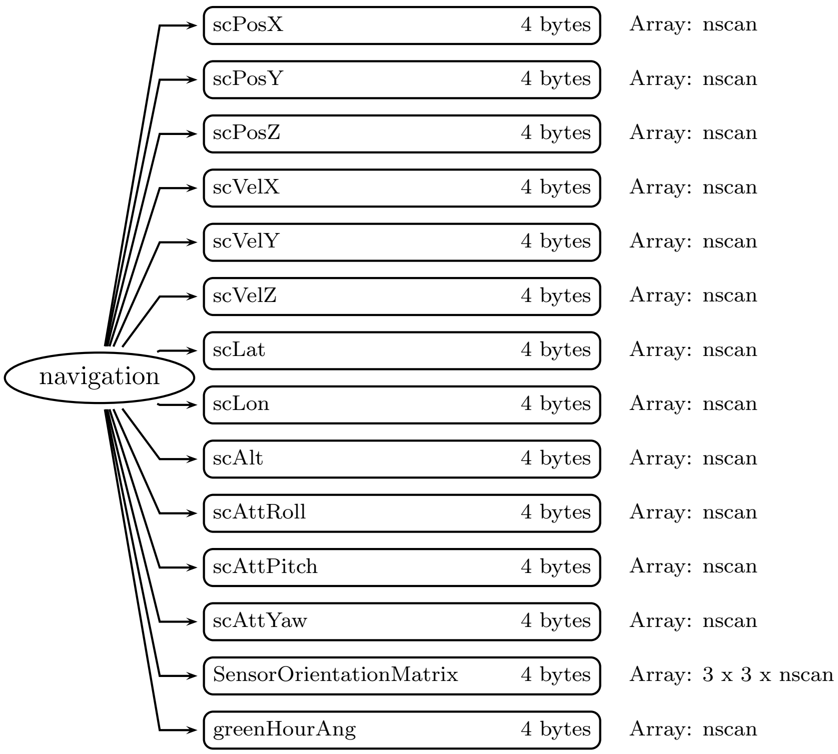

navigation
scPosX 4 bytes Array: nscan
scPosY 4 bytes Array: nscan
scPosZ 4 bytes Array: nscan
scVelX 4 bytes Array: nscan
scVelY 4 bytes Array: nscan
scVelZ 4 bytes Array: nscan
scLat 4 bytes Array: nscan
scLon 4 bytes Array: nscan
scAlt 4 bytes Array: nscan
scAttRoll 4 bytes Array: nscan
scAttPitch 4 bytes Array: nscan
scAttYaw 4 bytes Array: nscan
SensorOrientationMatrix 4 bytes Array: 3 x 3 x nscan
greenHourAng 4 bytes Array: nscan

Figure 5: Data Format Structure for 2A21, navigation

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

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.

sigmaZero (4-byte float, array size: nray x nscan):
The normalized surface cross section. Values range from -50 to 50 dB. Special values are defined as:

-9999.9	Missing value

pathAtten (4-byte float, array size: nray x nscan):
This is the best estimate of 2-way path integrated attenuation when rain is present and the reference is reliable or marginally reliable (see reliabFlag). This best estimate is a combination of various reference methods (see PIAweight). Values range from -50 to 50 dB. Special values are defined as:

-9999.9	Missing value

PIAalt (4-byte float, array size: refmethod x nray x nscan):
Alternate estimates of the 2-way path integrated attenuation when rain is present and the reference is reliable or marginally reliable (see RFactorAlt). The order in zero-based notation of the reference methods (refmethod) is:

```
0 - spatial, forward
1 - hybrid, forward (ocean only)
2 - spatial, backward
3 - hybrid, backward (ocean only)
4 - temporal (should be the same forward or backward)
```

Values range from -50 to 50 dB. Special values are defined as:

-9999.9	Missing value

PIAweight (4-byte float, array size: refmethod x nray x nscan):
A vector containing floating point weights indicating the combination of PIA estimates used to establish the best 2-way path integrated attenuation (pathAtten). Each value gives the relative weight used for each reference method. The order in zero-based notation of the reference methods (refmethod) is:

```
0 - spatial, forward
1 - hybrid, forward (ocean only)
2 - spatial, backward
3 - hybrid, backward (ocean only)
4 - temporal (should be the same forward or backward)
```

Values range from 0 to 1. Special values are defined as:

-9999.9	Missing value

reliabFlag (2-byte integer, array size: nray x nscan):
This flag indicates the reliability of the pathAtten estimate. Flag values are:

```
1 - PIA estimate is reliable
2 - PIA estimate is marginally reliable
3 - PIA estimate is unreliable
4 - PIA estimate is a lower bound to the path attenuation
9 - No PIA estimate, no-rain in ifov.
```

Values range from 1 to 9. Special values are defined as:
-9999 Missing value

reliabFactor (4-byte float, array size: nray x nscan):
The Reliability Factor is the ratio of the estimated value of path attenuation to the standard deviation associated with the mean value of the reference estimate. Values range from -10 to 10. Special values are defined as:
-9999.9 Missing value

RFactorAlt (4-byte float, array size: refmethod x nray x nscan):
Reliability factor for the alternate PIA estimates in PIAalt for each refmethod. The Reliability Factor is the ratio of the estimated value of path attenuation to the standard deviation associated with the mean value of the reference estimate. The order in zero-based notation of the reference methods (refmethod) is:

```
0 - spatial, forward
1 - hybrid, forward (ocean only)
2 - spatial, backward
3 - hybrid, backward (ocean only)
4 - temporal (should be the same forward or backward)
```

Values range from -10 to 10. Special values are defined as:
-9999.9 Missing value

rainFlag (2-byte integer, array size: nray x nscan):
The Rain Flag has the following values:

```
0: no rain;
1: rain present.
```

Values range from 0 to 1. Special values are defined as:
-9999 Missing value

incAngle (4-byte float, array size: nray x nscan):
The Incident Angle is the angle, in degrees, between the PR nadir and the radar beam. Values range from -30 to 30 degrees. Special values are defined as:
-9999.9 Missing value

refScanID (2-byte integer, array size: distance x direction x nray x nscan):
Provides scan information for the nearest and farthest reference points for the along-track methods. The values are (Current Scan) - (Reference Scan). These values are positive for the Forward estimates and negative for the Backward estimates. The dimensions in C notation are:

```
[0][0] - Forward - Near reference
[0][1] - Forward - Far reference
[1][0] - Backward - Near reference
[1][1] - Backward - Far reference
```

Values range from -9300 to 9300 scanNum. Special values are defined as:
-9999 Missing value

refMethodFlag (2-byte integer, array size: nray x nscan):
Flag for the reference method associated with the best PIA estimate (pathAtten).

```
        3 - insufficient number of data points
        4 - unknown background type
        5 - no-rain case and low SNR, no update to reference data
        9 - no rain case
```

Values range from 3 to 9. Special values are defined as:
-9999 Missing value

surfaceTracker (2-byte integer, array size: nray x nscan):
Status of surface tracker as a flag value.

```
        1 - surface tracker locked - central angle bin
        2 - surface tracker unlocked - central angle bin
        3 - peak surface return at normally sampled gate,
            outside central swath
        4 - peak surface return not at normally sampled gate,
            outside central swath
```

Values range from 1 to 4. Special values are defined as:
-9999 Missing value

surfTypeFlag (2-byte integer, array size: nray x nscan):
Flag value for the surface type for a given ifov. Flag values are:

```
        0 - Ocean
        1 - Land
        2 - Coast
        3 - Unknown or of a category other than those above or 'mixed'
```

Values range from 0 to 3. Special values are defined as:
-9999 Missing value

spare (4-byte float, array size: 5 x nray x nscan):
Contains developer output.

C Structure Header file:

```
#ifndef _TK_2A21_H_
#define _TK_2A21_H_
```

```
#ifndef _L2A21_NAVIGATION_
#define _L2A21_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;
} L2A21_NAVIGATION;

#endif

#ifndef _L2A21_SCANSTATUS_
#define _L2A21_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;
} L2A21_SCANSTATUS;

#endif

#ifndef _L2A21_SCANTIME_
```

```
#define _L2A21_SCANTIME_

typedef struct {
    short Year;
    signed char Month;
    signed char DayOfMonth;
    signed char Hour;
    signed char Minute;
    signed char Second;
    short MilliSecond;
    short DayOfYear;
} L2A21_SCANTIME;

#endif

#ifndef _L2A21_SWATH_
#define _L2A21_SWATH_

typedef struct {
    L2A21_SCANTIME ScanTime;
    double scanTime_sec;
    float Latitude[49];
    float Longitude[49];
    L2A21_SCANSTATUS scanStatus;
    L2A21_NAVIGATION navigation;
    float sigmaZero[49];
    float pathAtten[49];
    float PIAalt[49][5];
    float PIAweight[49][5];
    short reliabFlag[49];
    float reliabFactor[49];
    float RFactorAlt[49][5];
    short rainFlag[49];
    float incAngle[49];
    short refScanID[49][2][2];
    short refMethodFlag[49];
    short surfaceTracker[49];
    short surfTypeFlag[49];
    float spare[49][5];
} L2A21_SWATH;

#endif
```

```
#endif
```

Fortran Structure Header file:

```
STRUCTURE /L2A21_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 /L2A21_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 /L2A21_SCANTIME/
    INTEGER*2 Year
    BYTE Month
    BYTE DayOfMonth
    BYTE Hour
    BYTE Minute
    BYTE Second
    INTEGER*2 MilliSecond
    INTEGER*2 DayOfYear
END STRUCTURE

STRUCTURE /L2A21_SWATH/
    RECORD /L2A21_SCANTIME/ ScanTime
    REAL*8 scanTime_sec
    REAL*4 Latitude(49)
    REAL*4 Longitude(49)
    RECORD /L2A21_SCANSTATUS/ scanStatus
    RECORD /L2A21_NAVIGATION/ navigation
    REAL*4 sigmaZero(49)
    REAL*4 pathAtten(49)
    REAL*4 PIAalt(5,49)
    REAL*4 PIAweight(5,49)
    INTEGER*2 reliabFlag(49)
    REAL*4 reliabFactor(49)
    REAL*4 RFactorAlt(5,49)
    INTEGER*2 rainFlag(49)
    REAL*4 incAngle(49)
    INTEGER*2 refScanID(2,2,49)
    INTEGER*2 refMethodFlag(49)
    INTEGER*2 surfaceTracker(49)
    INTEGER*2 surfTypeFlag(49)
    REAL*4 spare(5,49)
END STRUCTURE
```