AATSR LEVEL 2 DETAILED PROCESSING MODEL & PARAMETER DATA LIST

SCIENCE AND TECHNOLOGY FACILITIES COUNCIL

RUTHERFORD APPLETON LABORATORY

Chilton, Didcot

Oxfordshire OX11 0QX

Prepared by:	A.R.Birks, RAL	Digitally signed by Andrew R Birks DN: cn=Andrew R Birks, o=STFC, ou, email=andrew.birks@stfc.ac.uk, c=GB Date: 2011.10.21 15:00:32 +01'00'	*Date:*
Approved by:	Caroline Cox, RAL	Caroline Cox 2011.10.21 15:08:24 +01'00'	*Date:*

Amendment Record

Issue	Revision	Date	Details
0	1	14 January 1997	Updated draft by RAL
0	2	12 February 1997	Revised by RAL
1	0	28 February 1997	First formal release
1	1	31 July 1997	Second Formal Release.
			Updated to match I/O DD Revision (Issue
			1 Revision 7). ECMWF Product added.
			Corrections and clarifications made.
1	2	21 November 1997	Updated and clarified in the light of
			comments received.
1	2	5 December 1997	Third formal release (to ESA).
1	3	26 January 1998	Corrected and clarified.
			AST Internal Variable Tables Revised
			Updated AST output assignments in
1	3		Module 16. Break Point Tables added.
		30 January 1998	Fourth formal release.
1	4	20th October 1998	Revised to take acount of SPRs to date.
1	4	11 March 1999	Revised to take account of SPRs to date.
			Definition of SQ ADS (ADS #0) added.
1	4	12 March 1999	Fifth formal release.
1	5	27 June 2000	Revised to take acount of SPRs to date.
1	5	27 June 2000	Revised to incorporate changes to the
			across-track banding scheme for SST
			derivation; to the infra-red brightness
			temperature precision in the ABT product,
			and to the definition of the Meteo product.
1	5	4 July 2000	Sixth formal release.
1	6	15 October 2001	LST algorithm added
1	6	2 June 2003	LST algorithm revised
1	6	18 June 2003	Seventh formal release
1	7	15 November 2005	LST Algorithm revised
			Typos corrected
1	7	2 June 2006	Eighth formal release
1	8	16 June 2011	Clarification of AST Confidence Flags
			Initialisation.
1	8	16 June 2011	Draft A Issued.
1	8	14th Oct. 2011	Ninth formal release

CONFIGURATION CONTROL

Document Last Updated by: **RAL on the 14th October 2011**

(to update these data, click on the field, hold down right mouse** button **and use Update Field)

Status(Draft / Controlled) : **Controlled**

TABLE OF CONTENTS

1 PURPOSE OF DOCUMENT

This document defines the Data Processing Model and the Parameter Data List for ENVISAT AATSR Level 2 processing.

2 INTRODUCTION

AATSR Level 2 processing encompasses the derivation of the GSST and ASST products from GBTR products output at Level 1B (ref. [1]). This document describes the step by step procedures which should be implemented within the ENVISAT Ground Segment processing to produce near real-time (NRT) and offline products.

2.1 Acronyms

AATSR	Advanced Along-Track Scanning Radiometer
ADS	Annotation Data Set; a data set within an ENVISAT product containing annotation data.
AST	Averaged Surface Temperature.
BB	Black Body
CAEID	Critical Algorithm Elements Identification Document
CAPA	Critical Analysis of Processing Algorithms Document
CFI	Customer Furnished Item
CGU	Clock Generation Unit
CRC	Cyclical Redundancy Check
CRRD	Computer Resource Requirements Document
DPM	Detailed Processing Model
DS	Data Set
DSD	Data Set Descriptor
DSR	Data Set Record
FODP	Flight Operation and Data Plan
FPA	Focal Plane Assembly
GBTR	Gridded Brightness Temperature/Reflectance
GOME	Global Ozone Monitoring Experiment
GSST	Gridded Sea Surface Temperature
I/O DD	Input/Output Data Definition Document
LST	Land Surface Temperature
LUT	Look Up Table
MDS	Measurement Data Set; a data set within an ENVISAT product containing instrument data
MPH	Main Product Header
MX BB	Minus X black body

NDVI Normalised Difference Vegetation Index
NRT Near Real Time

PAC Processing and Archiving Centre
PCSU Power Conditioning and Switching Unit
PDAS Payload Data Acquisition Station
PDHS Payload Data Handling Station
PDL Parameter Data List
PDS Payload Data Segment
PDP Prototype Development Plan
PRT Platinum Resistance Thermometer
PSM Pixel Selection Map
PX BB Plus X black body

SCC Stirling Cycle Cooler
SCP Signal Channel Processors
SPH Specific Product Header
SST Sea Surface Temperature

TDD Test Definition Document
TDS Test Data Sets
TPD Test Procedures Document
TMZ historical term for auxiliary data

VISCAL Visible calibration unit; on-board unit forming part of the AATSR instrument
for use in the calibration of the visible channels.

3 REFERENCE DOCUMENTS

Reference	Title	Number	Source
RD 1	AATSR Level 1B Detailed Processing Model & Parameter Data List	PO-TN-RAL-GS-10004	RAL
RD 2	AATSR Input / Output Data Definition Document	PO-TN-RAL-GS-10003	RAL
RD-3	Land Surface Temperature Measurement from Space: AATSR Algorithm Theoretical Basis Document	Fred Prata (CSIRO)	RAL or CSIRO
AD 1	ENVISAT-1 reference definitions document for	PO-TN-ESA-GS-00361	ESA

	mission related software		
AD 2	ENVISAT-1 Mission CFI Software: General Software User Manual	PO-IS-GMV-GS-0556	ESA

4 DETAILED PROCESSING MODEL

This section describes the level 2 processing. It includes a module by module breakdown of the processing structure. First a general overview of the processing structure can be found (see also Figure 4-1-1). This is followed by a more detailed description of each of the component modules; each module having:

- A functional description
- An interface definition
- An algorithm definition or detailed structure description
- A description of test procedures.

The interface definition consists of two tables, an Input table and an Internal table. The first deals principally with the interface to the IODD and to external files defined within the ENVISAT processing environment. The second defines parameters which are “internal to the processor”, that is to say parameters which are defined within a particular module, used by one or more modules. The Input table can contain parameters defined either in the IODD, or another external file - it represents the interface with the “outside world”. The Internal table contains newly defined parameters or parameters defined in a previously declared Internal table, which the module under discussion needs to access. Internal parameters can be local (to the module), or global. Global parameters are available to other modules, including the product output module. All global internal parameters are summarized in Table 5-1: Internal Parameter summary list.

Parameter names are defined using the following conventions:

IODD parameters: format: <ProductCode>-<Data set>-<ID number> (e.g. L0-MDS1-1)

Internal parameters format (global variables): <DPM>-INT-<SEQ> (e.g. L1B-INT-1), where SEQ is a unique number for any given DPM level.

The data tables have the following columns:

Parameter ID: refers to the ID in the IODD (for external parameters) or the internal data ID for internally generated parameters. These are also used within pseudo code and text descriptions to refer to the parameters. In the case of internal parameters, if this field is blank, or contains “local”, the variable is taken to be local to the module. For example, loop counters would generally be defined as local variables. If an Internal ID is defined, the variable is assumed to be global.

Variable: This is an optional entry, allowing a parameter to also be referred to within the text through a variable name, for ease of use, or to follow convention. If the column is blank, the parameter will always be referred to by its ID;

Name: A “long name” format, providing the means for a short description. This can be used for reference in text descriptions, but generally is not used within pseudo code for reasons of clarity.

Type: The parameter type, using standard ENVISAT PDS conventions;

Units: The parameter SI unit (where appropriate), and if necessary scaling factor;

Size: The space required by the parameter in bytes;

Fields: The number of fields for cases where the ID refers to an array of parameters;

Other conventions in use in this document are as follows:

Underscore is regarded as an alphabetic character in variable names.

Type font and style are of no significance; for example, the same variable is meant whether the name appears in italic or roman type, or in a different font.

Type of brackets is of no significance; either parentheses or square brackets masy be used equivalently. Parameter IDs are designed to enable cross-referencing between this document and the I/O DD, and between modules within this document, and they may be used as variable names to reduce ambiguity. Parameter IDs used in equations are generally enclosed in square brackets, to enable them to be subscripted. Thus if a parameter ID refers to an indexed or subscripted variable, the following notation may be used to associate the subscript ‘i’ with the ID: [L2-INT-nn](i).

For example

[L2-INT-101](i, j) is equivalent to I(ir12, n; i, j)

[L2-INT-110](i, j) is equivalent to frwrd_fill_state(i, j)

and so on.

Pointed brackets <> are (except for a few points in the Level 2 processing where they are used to denote averaged quantities: this should be clear from the context) metasyntactical; they enclose strings that are to be substituted by one of a set of optional strings to give the true variable name. For example, the construction <view>_fill_state(i, j) is to represent one of the two quantities nadir_fill_state(i, j) or frwrd_fill_state(i, j), according as whether the nadir or forward view data is being processed.more

Indices in equations may appear indifferently as subscripts or enclosed in brackets. Sometimes the convention of separating parenthesised indices with semicolons is used: e.g. *I(ch, v; i, j)*. The significance of this is that the indices preceding the square brackets are regarded as subscripts that may be thought of as part of the variable name (and therefore need not correspond to variables in an implementation) while those following the semicolon are array indices.

Indexing

The following indexing conventions are adopted generally:

i along track (image scan) index

j across track (image pixel) index (j = 0, 511)

Unless otherwise stated, indices start at zero.

For the purpose of indexing and identifying the AATSR channels, the following conventional numbering scheme will be adopted.

AATSR Channel	Symbol	Index (ch)
12 micron	ir12	1
11 micron	ir11	2
3.7 micron	ir37	3
1.6 micron	v16	4
0.870 micron	v870	5
0.670 micron	v670	6
0.55 micron	v555	7

Requirements are identified by numbers of the form (Req. <id>-<sequence>) where <id> is an identifier that is unique to the module or chapter, and <sequence> is the sequence number within the series identified by <id>.

4.1 Overview of Processing Structure

4.1.1 General

Figure 4-1-1 shows an overview of the Level 2 processing.

The main steps in the production of the Level 2 products are as follows:

- Derivation of Sea Surface Temperature (SST) and other parameters from the GBTR regridded brightness temperatures.
- Generation of averaged brightness temperatures and reflectances from the GBTR regridded brightness temperatures and visible channel reflectances.
- Derivation of averaged SST from the averaged brightness temperatures, and of NDVI from the averaged reflectances.

The processing makes use of some but not all of the supporting data from the GBTR ADS.

4.1.2 Input Annotation Data Sets (Module 1)

This module inputs those Annotation Data Sets of the GBTR product that are required for Level 2 Processing, and converts into appropriate units where necessary. It is described in Section 4.2.

4.1.3 Assemble Regridded Brightness Temperature Arrays (Module 2)

This module reads in grid co-ordinates and channel brightness temperatures / reflectances for forward and nadir views from the appropriate MDS of the GBTR product and arranges them in the required memory configuration. It is described in Section 4.3.

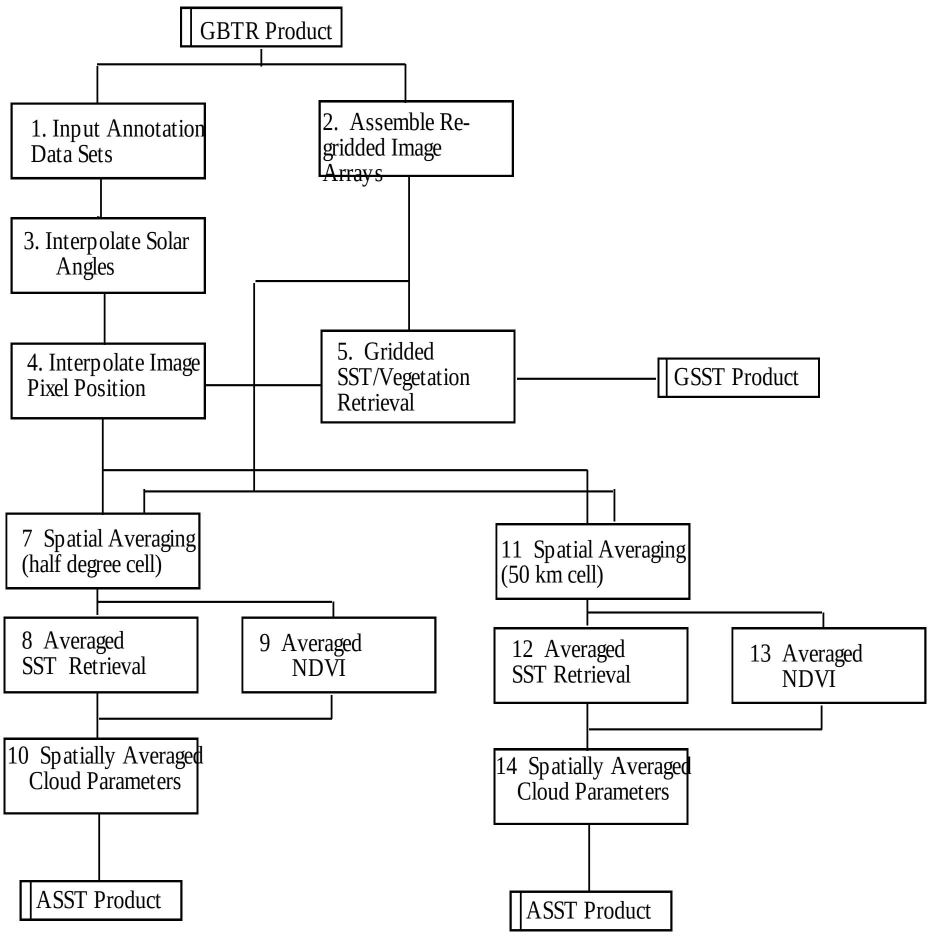

Figure 4-1-1. AATSR Level 2 Processing

4.1.4 Interpolate Solar Angles (Module 3)

The solar azimuth and elevation and satellite azimuth and elevation, all measured at the pixel, are available for a series of uniformly spaced tie point pixels in ADS #5 for the nadir view and in ADS #6 for the forward view images. The present module derives those angles that are required for level 2 processing at every scan, and at the mid-points of the bands, by linear interpolation, between these tie points. Only the solar elevation is required for Level 2 processing as presently defined. It is described is Section 4.4.

4.1.5 Interpolate Image Pixel position (Module 4)

The (geodetic) latitudes and longitudes of a series of uniformly space tie point pixels are available in ADS #3. This module derives the latitude and longitude of each of image pixel by linear interpolation, in two dimensions, between these tie points. The module is described in Section 4.5.

4.1.6 Gridded SST/Vegetation Retrieval (Module 5)

This module derives the contents of the GSST product at 1 km resolution from the infra-red brightness temperatures. It derives the sea surface temperature (SST) or, over land, the vegetation index (NDVI), at 1 Km resolution, using cloud free data. It is described in Section 4.6.

4.1.7 Output GSST Records (Module 6)

All data required for the GSST product is now available, and is formatted into the products described in the IODD. The module is described in Section 4.16.

4.1.8 Spatial Averaging (Half-Degree Cell) (Module 7)

For the averaged products in half-degree cells, the globe is imagined as divided into cells 0.5° in latitude by 0.5° in longitude, and these cells are further subdivided into 9 sub-cells extending 10 arcmin in latitude by 10 arcmin in longitude. For each channel, the average brightness temperature (for the infra-red channels) or reflectance (for the visible channels) is averaged over all pixels of each type that fall within each sub-cell, to give distributions of a brightness temperature and radiance at 10 arc minute resolution. Averages are performed for the forward and nadir views separately, and a separate average is performed for each surface type (land and sea) and cloud state (clear or cloudy). There are thus 4 averages per channel per view. The mean across-track band number in each cell is also derived, for use by the averaged SST algorithm. The module is described in Section 4.8.

4.1.9 Averaged SST Retrieval (Half-Degree Cell) (Module 8)

This module derives the averaged SST for the cells and sub-cells from the averaged brightness temperatures determined above, for cells containing sea. It is described in Section 4.9.

4.1.10 Averaged NDVI Retrieval (Half-Degree Cell) (Module 9)

The NDVI is calculated for each sub-cell for which average reflectances over land have been calculated. The averaged NDVI over all the subcells, and its standard deviation, are also computed. The module is described in Section 4.10.

4.1.11 Spatially Averaged Cloud parameters (Half-Degree Cell) (Module 10)

This module provides physical information on the cloud state additional to the results of the cloud flagging provided in the Level 1b product. The product is based on the same half-degree cells defined above. In particular it derives an estimate of the cloud-top temperature. The latter is interpreted as the mean brightness temperature of the coldest 25% of the cloudy pixels in the cell. The module is described in Section 4.11.

4.1.12 Spatial Averaging (50 km cell) (Module 11)

This module derives spatially averaged brightness temperatures and reflectances as in Module 7, but averaged over cells and subcells of nominal dimensions 50 km x 50 km, and 17 x 17 km, respectively. It is described in Section 4.12

4.1.13 Averaged SST Retrieval (50 km cell) (Module 12)

This module derives spatially averaged brightness temperatures and reflectances as in Module 8, but averaged over cells and subcells of nominal dimensions 50 km x 50 km, and 17 x 17 km, respectively. It is described in Section 4.13

4.1.14 Averaged NDVI Retrieval (50 km cell) (Module 13)

This module derives spatially averaged NDVI as in Module 9, but averaged over cells and subcells of nominal dimensions 50 km x 50 km, and 17 x 17 km, respectively. It is described in Section 4.14.

4.1.15 Spatially Averaged Cloud parameters (50 km cell) (Module 14)

This module derives cloud parameters as in Module 9, but based on a cell of nominal dimensions 50 km x 50 km. It is described in Section 4.15.

4.1.16 Output AST Records (Module 15)

All data required for the AST product is now available, and is formatted into the products described in the IODD. The module is described in Section 4.16.

4.1.17 Output ECMWF Product (Module 16)

The ECMWF Averaged SST Product consists of an additional extraction of the AST product Measurement Data Set MDS #3. The product is generated in this module, which is described in Section 4.17.

4.1.18 Breakpoints

The following data shall be used as breakpoints in the testing of the Level 2 process.

Interpolated Solar Angles at the output of Module 3.

Interpolated pixel co-ordinates at the output of Module 4.

GSST Product Outputs from module 5.

AST product outputs from modules 7 - 15.

The table below indicates the accuracy with which the data should be verified against the output of the reference processor.

Parameter ID	Name	Verificaion Accuracy
	From Module 3 (Interpolate Solar Angles)	
L2-INT-120	nadir_band_edge_solar_elevation(i, k)	1 part in 1e6
L2-INT-140	frwrd_band_edge_solar_elevation(i, k)	1 part in 1e6
L2-INT-124	nadir_band_centre_solar_elevation(i, k′)	1 part in 1e6
L2-INT-144	frwrd_band_centre_solar_elevation(i, k′)	1 part in 1e6
	From Module 4 (Interpolate Image Pixel Position)	
L2-INT-160	image latitude	1 part in 1e6
L2-INT-161	image longitude	1 part in 1e6
	From Module 5 (Gridded SST/Vegetation Retrieval)	

L2-INT-270	nadir_image_field(i, j)	0.01K
L2-INT-271	combined_image_field(i, j)	0.01 K or 1 lsb
L2-INT-272	gsst_confidence_word(i, j)	Generally exact
	From Modules 7 - 15	
	AST parameters as tabulated below	See Tables below

Table 4-2-1. Level 2 Breakpoints

Note: In the table above, 'Generally exact' relates to flags or quantities of type integer, and indicates that test results should agree exactly with the reference processor in the majority of cases, but that a small number (TBD) of discepancies may acceptable owing to differences in machine precision.

The following tables describe the formats specified for the breakpoint outputs.

Table 4-2-2: Break Point #1 Record: nadir solar and viewing angles

Parameter ID	Start byte	End byte	Field Description	Type	Units	Field size	Fields
none	0	3	image row index (i)	sl	n/a	4	1
L2-INT-120	4	47	nadir_band_edge_solar_elevation(i, k)	float	degrees	4	11
L2-INT-124	48	87	nadir_band_centre_solar_elevation(i, k′)	float	degrees	4	10
L2-INT-140	88	131	frwrd_band_edge_solar_elevation(i, k)	float	degrees	4	11
L2-INT-144	132	171	frwrd_band_centre_solar_elevation(i, k′)	float	degrees	4	10

Table 4-2-3: Break Point #2 Record: nadir view instrument pixel numbers

Parameter ID	Start byte	End byte	Field Description	Type	Units	Field size	Fields
none	0	3	image row index (i)	sl	n/a	4	1
L2-INT-160	4	2051	image_latitude(i, j)	float	degrees	4	512
L2-INT-161	2052	4099	image_longitude(i, j)	float	degrees	4	512

Table 4-2-4: Break Point #3 Record: Gridded product record

Parameter ID	Start byte	End byte	Field Description	Type	Units	Field Size	Fields	Verification Accuracy
none	0	3	image row index (i)	sl	n/a	4	1	exact
L2-INT-272	4	1027	confidence words	us	flags	2	512	generally exact
L2-INT-270	1028	2051	nadir field prior to SST smoothing	ss	K/100	2	512	0.01 K
L2-INT-271	2052	3075	combined field prior to SST smoothing	ss	K/100	2	512	0.01 K

Note 1. The GSST product is switchable, so that the contents of these MDS fields depend on the setting of the cloud land flags. The confidence word is included in the above records so that the product is interpretable. The units of these quantities depend on the flag settings; values quoted are for cloud-free sea data.

Table 4-2-5: Break point #4 Record, AST Sea Record, 30 arc minute cell

Field No.	Start byte	End byte	View	Field Description	Type	Units	Field Size	Fields	Verification Accuracy
L2-INT-30	0	7	n/a	cell UTC	double	days	8	1	
L2-INT-47	8	11	n/a	cell latitude	sl	μdeg	4	1	100 μdeg
L2-INT-48	12	15	n/a	cell longitude	sl	μdeg	4	1	100 μdeg
L2-INT-344	16	19	N	total of clear sea pixels, nadir view	sl	none	4	1	exact
L2-INT-345	20	23	F	total of clear sea pixels, forward view	sl	none	4	1	exact
L2-INT-346	24	27	N	total of cloudy sea pixels, nadir view	sl	none	4	1	exact

L2-INT-347	28	31	F	total of cloudy sea pixels, forward view	sl	none	4	1	exact
L2-INT-348	32	2031	N	nadir histogram (sea cell)	ss	none	2	1000	exact
L2-INT-349	2032	4031	F	forward histogram (sea cell)	ss	none	2	1000	exact
L2-INT-40	4032	4045	N	Total clear pixels over sea in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-40	4046	4059	N	Total cloudy pixels over sea in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-40	4060	4073	F	Total clear pixels over sea in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-40	4074	4087	F	Total cloudy pixels over sea in cell, ch = 1, ...7	ss	none	2	7	exact
AST-MDS16-73	4088	4089	N	Lowest 11 micron BT of all cloudy pixels, nadir view	ss	K/100	2	1	0.01 K
AST-MDS16-74	4090	4091	N	Corresponding 12 micron BT, nadir view	ss	K/100	2	1	0.01 K
AST-MDS16-75	4092	4092	N	Corresponding 3.7 micron BT, nadir view	ss	K/100	2	1	0.01 K
AST-MDS16-76	4094	4094	N	Corresponding 1.6 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS16-77	4096	4096	N	Corresponding 0.87 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS16-78	4098	4098	N	Corresponding 0.67 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS16-79	4100	4101	N	Corresponding 0.55 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS16-80	4102	4103	F	Lowest 11 micron BT of all cloudy pixels, forward view	ss	K/100	2	1	0.01 K
AST-MDS16-81	4104	4105	F	Corresponding 12 micron BT, forward view	ss	K/100	2	1	0.01 K
AST-MDS16-82	4106	4107	F	Corresponding 3.7 micron BT, forward view	ss	K/100	2	1	0.01 K
AST-MDS16-83	4108	4109	F	Corresponding 1.6 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS16-84	4110	4111	F	Corresponding 0.87 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS16-85	4112	4113	F	Corresponding 0.67 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS16-86	4114	4115	F	Corresponding 0.55 micron reflectance, forward view	ss	%/100	2	1	0.01 %

Table 4-2-6: Break point #5 Record, AST Land Record, 30 arc minute cell

Field No.	Start byte	End byte	View	Field Description	Type	Units	Field Size	Fields	Verification Accuracy
L2-INT-30	0	7	n/a	cell UTC	double	days	8	1	
L2-INT-47	8	11	n/a	cell latitude	sl	μdeg	4	1	100 μdeg
L2-INT-48	12	15	n/a	cell longitude	sl	μdeg	4	1	100 μdeg
L2-INT-328	16	19	N	total of clear land pixels, nadir view	sl	none	4	1	exact
L2-INT-329	20	23	F	total of clear land pixels, forward view	sl	none	4	1	exact
L2-INT-330	24	27	N	total of cloudy land pixels, nadir view	sl	none	4	1	exact
L2-INT-331	28	31	F	total of cloudy land pixels, forward view	sl	none	4	1	exact
L2-INT-332	32	2031	N	nadir histogram (land cell)	ss	none	2	1000	exact
L2-INT-333	2032	4031	F	forward histogram (land cell)	ss	none	2	1000	exact
L2-INT-40	4032	4045	N	Total clear pixels over land in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-40	4046	4059	N	Total cloudy pixels over land in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-40	4060	4073	F	Total clear pixels over land in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-40	4074	4087	F	Total cloudy pixels over land in cell, ch = 1, ...7	ss	none	2	7	exact
AST-MDS12-77	4088	4089	N	Lowest 11 micron BT of all cloudy pixels, nadir view	ss	K/100	2	1	0.01 K
AST-MDS12-78	4090	4091	N	Corresponding 12 micron BT, nadir view	ss	K/100	2	1	0.01 K
AST-MDS12-79	4092	4092	N	Corresponding 3.7 micron BT, nadir view	ss	K/100	2	1	0.01 K
AST-MDS12-80	4094	4094	N	Corresponding 1.6 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS12-81	4096	4096	N	Corresponding 0.87 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS12-82	4098	4098	N	Corresponding 0.67 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS12-83	4100	4101	N	Corresponding 0.55 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS12-84	4102	4103	F	Lowest 11 micron BT of all cloudy pixels, forward view	ss	K/100	2	1	0.01 K
AST-MDS12-85	4104	4105	F	Corresponding 12 micron BT, forward view	ss	K/100	2	1	0.01 K
AST-MDS12-86	4106	4107	F	Corresponding 3.7 micron BT, forward view	ss	K/100	2	1	0.01 K
AST-MDS12-87	4108	4109	F	Corresponding 1.6 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS12-88	4110	4111	F	Corresponding 0.87 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS12-89	4112	4113	F	Corresponding 0.67 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS12-90	4114	4115	F	Corresponding 0.55 micron reflectance, forward view	ss	%/100	2	1	0.01 %

Table 4-2-7: Break point #6 Record, AST Sea Record, 10 arc minute cell

Field No.	Start byte	End byte	View	Field Description	Type	Units	Field Size	Fields	Verification Accuracy
L2-INT-31	0	11	n/a	utc(k, cell):sub-cell UTC	double	days	8	1	
L2-INT-49	12	12	n/a	nadir view day/night flag	ss	flag	2	1	exact
L2-INT-50	13	15	n/a	forward view day/night flag	ss	flag	2	1	exact
L2-INT-45	16	19	n/a	nadir solar elevation for sub-cell	float	degrees	4	1	1 part in 1e6
L2-INT-46	20	23	n/a	frwrd solar elevation for sub-cell	float	degrees	4	1	1 part in 1e6
L2-INT-32	24	25	n/a	sub-cell latitude	sl	µdeg	4	1	100 µdeg
L2-INT-33	26	27	N	sub-cell longitude	sl	µdeg	4	1	100 µdeg
L2-INT-34	28	29	N	sub-cell across-track band	ss	none	2	1	exact
L2-INT-36	30	31	N	sub-cell total, ch = 1, ..., 7, clear pixels, nadir view	sl	n/a	4	7	500 lsb
L2-INT-36	58	85	N	sub-cell total, ch = 1, ..., 7, cloudy pixels, nadir view	sl	n/a	4	7	500 lsb
L2-INT-36	86	113	F	sub-cell total, ch = 1, ..., 7, clear pixels, forward view	sl	n/a	4	7	500 lsb
L2-INT-36	114	141	F	sub-cell total, ch = 1, ..., 7, cloudy pixels, frwrd view	sl	n/a	4	7	500 lsb
L2-INT-37	142	155	N	sub-cell valid pixel count, ch = 1, ..., 7, clear pixels	ss	none	2	7	exact
L2-INT-37	156	169	N	sub-cell valid pixel count, ch = 1, ..., 7, cloudy pixels	ss	none	2	7	exact
L2-INT-37	170	183	F	sub-cell valid pixel count, ch = 1, ..., 7, clear pixels	ss	none	2	7	exact
L2-INT-37	184	197	F	sub-cell valid pixel count, ch = 1, ..., 7, cloudy pixels	ss	none	2	7	exact
L2-INT-355	198	199	N	sub-cell filled pixel count, clear pixels, nadir view	ss	none	2	1	exact
L2-INT-355	200	201	N	sub-cell filled pixel count, cloudy pixels, nadir view	ss	none	2	1	exact
L2-INT-355	202	203	F	sub-cell filled pixel count, clear pixels, forward view	ss	none	2	1	exact
L2-INT-355	204	205	F	sub-cell filled pixel count, cloudy pixels, frwrd view	ss	none	2	1	exact
L2-INT-356	206	209	N	cumulative across-track band sum	sl	none	4	1	1 lsb
L2-INT-357	210	211	N	mean across-track band number	ss	none	2	1	1 lsb

Table 4-2-8: Break point #7 Record, AST Land, 10 arc minute cell

Field No.	Start byte	End byte	View	Field Description	Type	Units	Field Size	Fields	Verification Accuracy
L2-INT-31	0	7	n/a	utc(k, cell):sub-cell UTC	double	days	8	1	
L2-INT-49	8	9	N	nadir view day/night flag	ss	flag	2	1	exact
L2-INT-50	10	11	F	forward view day/night flag	ss	flag	2	1	exact
L2-INT-45	12	15	N	nadir solar elevation for sub-cell	float	degrees	4	1	1 part in 1e6
L2-INT-46	16	19	F	frwrd solar elevation for sub-cell	float	degrees	4	1	1 part in 1e6
L2-INT-32	20	23	n/a	sub-cell latitude	sl	µdeg	4	1	100 µdeg
L2-INT-33	24	27	n/a	sub-cell longitude	sl	µdeg	4	1	100 µdeg
L2-INT-34	28	29	n/a	sub-cell across-track band	ss	none	2	1	exact
L2-INT-36	30	57	N	sub-cell total, ch = 1, ..., 7, clear pixels, nadir view	sl	n/a	4	7	500 lsb
L2-INT-36	58	85	N	sub-cell total, ch = 1, ..., 7, cloudy pixels, nadir view	sl	n/a	4	7	500 lsb
L2-INT-36	86	113	F	sub-cell total, ch = 1, ..., 7, clear pixels, forward view	sl	n/a	4	7	500 lsb
L2-INT-36	114	141	F	sub-cell total, ch = 1, ..., 7, cloudy pixels, frwrd view	sl	n/a	4	7	500 lsb
L2-INT-37	142	155	N	sub-cell valid pixel count, ch = 1, ..., 7, clear pixels	ss	none	2	7	exact
L2-INT-37	156	169	N	sub-cell valid pixel count, ch = 1, ..., 7, cloudy pixels	ss	none	2	7	exact
L2-INT-37	170	183	F	sub-cell valid pixel count, ch = 1, ..., 7, clear pixels	ss	none	2	7	exact
L2-INT-37	184	197	F	sub-cell valid pixel count, ch = 1, ..., 7, cloudy pixels	ss	none	2	7	exact
L2-INT-355	198	199	N	sub-cell filled pixel count, clear pixels, nadir view	ss	none	2	1	exact
L2-INT-355	200	201	N	sub-cell filled pixel count, cloudy pixels, nadir view	ss	none	2	1	exact
L2-INT-355	202	203	F	sub-cell filled pixel count, clear pixels, forward view	ss	none	2	1	exact

L2-INT-355	204	205	F	sub-cell filled pixel count, cloudy pixels, frwrd view	ss	none	2	1	exact
L2-INT-356	206	209	N	cumulative across-track band sum	sl	none	4	1	1 lsb
L2-INT-357	210	211	N	mean across-track band number	ss	none	2	1	1 lsb

Table 4-2-9: Break point #8 Record, AST Sea Record, 50 km cell

Field No.	Start byte	End byte	View	Field Description	Type	Units	Field Size	Fields	Verification Accuracy
L2-INT-20	0	7	n/a	cell UTC	double	days	8	1	
L2-INT-77	8	11	n/a	cell latitude	sl	μdeg	4	1	100 μdeg
L2-INT-78	12	15	n/a	cell longitude	sl	μdeg	4	1	100 μdeg
L2-INT-444	16	19	N	total of clear sea pixels, nadir view	sl	none	4	1	exact
L2-INT-445	20	23	F	total of clear sea pixels, forward view	sl	none	4	1	exact
L2-INT-446	24	27	N	total of cloudy sea pixels, nadir view	sl	none	4	1	exact
L2-INT-447	28	31	F	total of cloudy sea pixels, forward view	sl	none	4	1	exact
L2-INT-448	32	2031	N	nadir histogram (sea cell)	ss	none	2	1000	exact
L2-INT-449	2032	4031	F	forward histogram (sea cell)	ss	none	2	1000	exact
L2-INT-70	4032	4045	N	Total clear pixels over sea in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-70	4046	4059	N	Total cloudy pixels over sea in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-70	4060	4073	F	Total clear pixels over sea in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-70	4074	4087	F	Total cloudy pixels over sea in cell, ch = 1, ...7	ss	none	2	7	exact
AST-MDS13-73	4088	4089	N	Lowest 11 micron BT of all cloudy pixels, nadir view	ss	K/100	2	1	0.01 K
AST-MDS13-74	4090	4091	N	Corresponding 12 micron BT, nadir view	ss	K/100	2	1	0.01 K
AST-MDS13-75	4092	4092	N	Corresponding 3.7 micron BT, nadir view	ss	K/100	2	1	0.01 K
AST-MDS13-76	4094	4094	N	Corresponding 1.6 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS13-77	4096	4096	N	Corresponding 0.87 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS13-78	4098	4098	N	Corresponding 0.67 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS13-79	4100	4101	N	Corresponding 0.55 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS13-80	4102	4103	F	Lowest 11 micron BT of all cloudy pixels, forward view	ss	K/100	2	1	0.01 K
AST-MDS13-81	4104	4105	F	Corresponding 12 micron BT, forward view	ss	K/100	2	1	0.01 K
AST-MDS13-82	4106	4107	F	Corresponding 3.7 micron BT, forward view	ss	K/100	2	1	0.01 K
AST-MDS13-83	4108	4109	F	Corresponding 1.6 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS13-84	4110	4111	F	Corresponding 0.87 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS13-85	4112	4113	F	Corresponding 0.67 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS13-86	4114	4115	F	Corresponding 0.55 micron reflectance, forward view	ss	%/100	2	1	0.01 %

Table 4-2-10: Break point #9 Record, AST Land Record, 50 km cell

Field No.	Start byte	End byte	View	Field Description	Type	Units	Field Size	Fields	Verification Accuracy
L2-INT-20	0	7	n/a	cell UTC	double	days	8	1	
L2-INT-77	8	11	n/a	cell latitude	sl	μdeg	4	1	100 μdeg
L2-INT-78	12	15	n/a	cell longitude	sl	μdeg	4	1	100 μdeg
L2-INT-428	16	19	N	total of clear land pixels, nadir view	sl	none	4	1	exact
L2-INT-429	20	23	F	total of clear land pixels, forward view	sl	none	4	1	exact
L2-INT-430	24	27	N	total of cloudy land pixels, nadir view	sl	none	4	1	exact
L2-INT-431	28	31	F	total of cloudy land pixels, forward view	sl	none	4	1	exact
L2-INT-432	32	2031	N	nadir histogram (land cell)	ss	none	2	1000	exact
L2-INT-433	2032	4031	F	forward histogram (land cell)	ss	none	2	1000	exact
L2-INT-70	4032	4045	N	Total clear pixels over land in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-70	4046	4059	N	Total cloudy pixels over land in cell, ch = 1, ...7	ss	none	2	7	exact
L2-INT-70	4060	4073	F	Total clear pixels over land in cell, ch = 1, ...7	ss	none	2	7	exact

L2-INT-70	4074	4087	F	Total cloudy pixels over land in cell, ch = 1, ...7	ss	none	2	7	exact
AST-MDS9-77	4088	4089	N	Lowest 11 micron BT of all cloudy pixels, nadir view	ss	K/100	2	1	0.01 K
AST-MDS9-78	4090	4091	N	Corresponding 12 micron BT, nadir view	ss	K/100	2	1	0.01 K
AST-MDS9-79	4092	4092	N	Corresponding 3.7 micron BT, nadir view	ss	K/100	2	1	0.01 K
AST-MDS9-80	4094	4094	N	Corresponding 1.6 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS9-81	4096	4096	N	Corresponding 0.87 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS9-82	4098	4098	N	Corresponding 0.67 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS9-83	4100	4101	N	Corresponding 0.55 micron reflectance, nadir view	ss	%/100	2	1	0.01 %
AST-MDS9-84	4102	4103	F	Lowest 11 micron BT of all cloudy pixels, forward view	ss	K/100	2	1	0.01 K
AST-MDS9-85	4104	4105	F	Corresponding 12 micron BT, forward view	ss	K/100	2	1	0.01 K
AST-MDS9-86	4106	4107	F	Corresponding 3.7 micron BT, forward view	ss	K/100	2	1	0.01 K
AST-MDS9-87	4108	4109	F	Corresponding 1.6 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS9-88	4110	4111	F	Corresponding 0.87 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS9-89	4112	4113	F	Corresponding 0.67 micron reflectance, forward view	ss	%/100	2	1	0.01 %
AST-MDS9-90	4114	4115	F	Corresponding 0.55 micron reflectance, forward view	ss	%/100	2	1	0.01 %

Table 4-2-11: Break point #10 Record, AST Sea Record, 17 km cell

Field No.	Start byte	End byte	View	Field Description	Type	Units	Field Size	Fields	Verification Accuracy
L2-INT-21	0	11	n/a	utc(k, cell):sub-cell UTC	double	days	8	1	
L2-INT-79	12	12	n/a	nadir view day/night flag	ss	flag	2	1	exact
L2-INT-80	13	15	n/a	forward view day/night flag	ss	flag	2	1	exact
L2-INT-75	16	19	n/a	nadir solar elevation for sub-cell	float	degrees	4	1	1 part in 1e6
L2-INT-76	20	23	n/a	frwrd solar elevation for sub-cell	float	degrees	4	1	1 part in 1e6
L2-INT-62	24	25	n/a	sub-cell latitude	sl	µdeg	4	1	100 µdeg
L2-INT-63	26	27	N	sub-cell longitude	sl	µdeg	4	1	100 µdeg
L2-INT-64	28	29	N	sub-cell across-track band	ss	none	2	1	exact
L2-INT-66	30	31	N	sub-cell total, ch = 1, ..., 7, clear pixels, nadir view	sl	n/a	4	7	500 lsb
L2-INT-66	58	85	N	sub-cell total, ch = 1, ..., 7, cloudy pixels, nadir view	sl	n/a	4	7	500 lsb
L2-INT-66	86	113	F	sub-cell total, ch = 1, ..., 7, clear pixels, forward view	sl	n/a	4	7	500 lsb
L2-INT-66	114	141	F	sub-cell total, ch = 1, ..., 7, cloudy pixels, frwrd view	sl	n/a	4	7	500 lsb
L2-INT-67	142	155	N	sub-cell valid pixel count, ch = 1, ..., 7, clear pixels	ss	none	2	7	exact
L2-INT-67	156	169	N	sub-cell valid pixel count, ch = 1, ..., 7, cloudy pixels	ss	none	2	7	exact
L2-INT-67	170	183	F	sub-cell valid pixel count, ch = 1, ..., 7, clear pixels	ss	none	2	7	exact
L2-INT-67	184	197	F	sub-cell valid pixel count, ch = 1, ..., 7, cloudy pixels	ss	none	2	7	exact
L2-INT-455	198	199	N	sub-cell filled pixel count, clear pixels, nadir view	ss	none	2	1	exact
L2-INT-455	200	201	N	sub-cell filled pixel count, cloudy pixels, nadir view	ss	none	2	1	exact
L2-INT-455	202	203	F	sub-cell filled pixel count, clear pixels, forward view	ss	none	2	1	exact
L2-INT-455	204	205	F	sub-cell filled pixel count, cloudy pixels, frwrd view	ss	none	2	1	exact
L2-INT-456	206	209	N	cumulative across-track band sum	sl	none	4	1	1 lsb
L2-INT-457	210	211	N	mean across-track band number	ss	none	2	1	1 lsb

Table 4-2-12: Break point #11 Record, AST Land, 17 km cell

Field No.	Start byte	End byte	View	Field Description	Type	Units	Field Size	Fields	Verification Accuracy
L2-INT-21	0	7	n/a	utc(k, cell):sub-cell UTC	double	days	8	1	
L2-INT-79	8	9	N	nadir view day/night flag	ss	flag	2	1	exact
L2-INT-80	10	11	F	forward view day/night flag	ss	flag	2	1	exact

L2-INT-75	12	15	N	nadir solar elevation for sub-cell	float	degrees	4	1	1 part in 1e6
L2-INT-76	16	19	F	frwrd solar elevation for sub-cell	float	degrees	4	1	1 part in 1e6
L2-INT-62	20	23	n/a	sub-cell latitude	sl	μdeg	4	1	100 μdeg
L2-INT-63	24	27	n/a	sub-cell longitude	sl	μdeg	4	1	100 μdeg
L2-INT-64	28	29	n/a	sub-cell across-track band	ss	none	2	1	exact
L2-INT-66	30	57	N	sub-cell total, ch = 1, ..., 7, clear pixels, nadir view	sl	n/a	4	7	500 lsb
L2-INT-66	58	85	N	sub-cell total, ch = 1, ..., 7, cloudy pixels, nadir view	sl	n/a	4	7	500 lsb
L2-INT-66	86	113	F	sub-cell total, ch = 1, ..., 7, clear pixels, forward view	sl	n/a	4	7	500 lsb
L2-INT-66	114	141	F	sub-cell total, ch = 1, ..., 7, cloudy pixels, frwrd view	sl	n/a	4	7	500 lsb
L2-INT-67	142	155	N	sub-cell valid pixel count, ch = 1, ..., 7, clear pixels	ss	none	2	7	exact
L2-INT-67	156	169	N	sub-cell valid pixel count, ch = 1, ..., 7, cloudy pixels	ss	none	2	7	exact
L2-INT-67	170	183	F	sub-cell valid pixel count, ch = 1, ..., 7, clear pixels	ss	none	2	7	exact
L2-INT-67	184	197	F	sub-cell valid pixel count, ch = 1, ..., 7, cloudy pixels	ss	none	2	7	exact
L2-INT-455	198	199	N	sub-cell filled pixel count, clear pixels, nadir view	ss	none	2	1	exact
L2-INT-455	200	201	N	sub-cell filled pixel count, cloudy pixels, nadir view	ss	none	2	1	exact
L2-INT-455	202	203	F	sub-cell filled pixel count, clear pixels, forward view	ss	none	2	1	exact
L2-INT-455	204	205	F	sub-cell filled pixel count, cloudy pixels, frwrd view	ss	none	2	1	exact
L2-INT-456	206	209	N	cumulative across-track band sum	sl	none	4	1	1 lsb
L2-INT-457	210	211	N	mean across-track band number	ss	none	2	1	1 lsb

4.2 Module Definition: Input Annotation Data Sets

4.2.1 Functional Description

The Annotation Data Sets of the GBTR product that are required for Level 2 Processing are read into memory, and converted into appropriate units where necessary. Data are required from data sets ADS #3 (grid pixel latitude and longitude), and ADS #5, #6 (solar angles). In addition data sets ADS #1, ADS #2 and ADS #4 are needed for AST record time tagging.

4.2.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
GBTR-ADS1-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS1-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS1-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS1-4		image scan y coordinate	sl	m	4	1
GBTR-ADS1-5		instrument scan number, nadir view	us	none	2	512
GBTR-ADS1-6		pixel number, nadir view	us	none	2	512
GBTR-ADS2-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS2-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS2-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS2-4		image scan y coordinate	sl	m	4	1
GBTR-ADS2-5		instrument scan number, forward view	us	none	2	512
GBTR-ADS2-6		pixel number, forward view	us	none	2	512
GBTR-ADS3-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS3-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS3-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS3-4		image scan y coordinate	sl	m	4	1
GBTR-ADS3-5		tie point latitudes	sl	μdeg	4	23
GBTR-ADS3-6		tie point longitudes	sl	μdeg	4	23

GBTR-ADS3-7		latitude corrections, nadir view	sl	µdeg	4	23
GBTR-ADS3-8		longitude corrections, nadir view	sl	µdeg	4	23
GBTR-ADS3-9		latitude corrections, forward view	sl	µdeg	4	23
GBTR-ADS3-10		longitude corrections, forward view	sl	µdeg	4	23
GBTR-ADS3-11		Topographic Altitude	ss	metres	2	23
GBTR-ADS4-1		Scan UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS4-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS4-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS4-4		instrument scan number	us	none	2	1
GBTR-ADS4-5		tie pixel x coordinate	sl	m	4	94
GBTR-ADS4-6		tie pixel y coordinate	sl	m	4	94
GBTR-ADS5-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS5-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS5-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS5-4		image scan y coordinate	sl	m	4	1
GBTR-ADS5-5		tie point solar elevation, nadir view	sl	mdeg	4	11
GBTR-ADS5-6		tie point satellite elevation, nadir view	sl	mdeg	4	11
GBTR-ADS5-7		tie point solar azimuth, nadir view	sl	mdeg	4	11
GBTR-ADS5-8		tie point satellite azimuth, nadir view	sl	mdeg	4	11
GBTR-ADS6-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS6-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS6-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS6-4		image scan y coordinate	sl	m	4	1
GBTR-ADS6-5		tie point solar elevation, forward view	sl	mdeg	4	11
GBTR-ADS6-6		tie point satellite elevation, forward view	sl	mdeg	4	11
GBTR-ADS6-7		tie point solar azimuth, forward view	sl	mdeg	4	11
GBTR-ADS6-8		tie point satellite azimuth, forward view	sl	mdeg	4	11

Table 4-2-1: Input Data Table - Input Annotation Data Sets

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-INT-18	MJDT(4)	Scan UTC in MJD Format	4*sl	MJD	16	per sg
L2-INT-23	MJDP[0]/(1)	Scan UTC in processing format	double	days	8	
L2-INT-24	MJDP[1]/(2)	Scan delta UT1 (dummy)	double		8	
L2-INT-26	time(sg)	scan UTC	double	days	8	per sg
L2-INT-27	scan(sg)	scan number corresponding to time(sg)	us	none	2	per sg
L2-INT-134	scn_nadir(ig, j)	nadir view instrument scan number	us	none	4	j = 0, 511
L2-INT-135	pxl_nadir(ig, j)	nadir view instrument pixel number	us	none	4	j = 0, 511
L2-INT-154	scn_frwrd(ig, j)	forward view instrument scan number	us	none	4	j = 0, 511
L2-INT-155	pxl_frwrd(ig, j)	forward view instrument pixel number	us	none	4	j = 0, 511
local	sg	index to instrument scan granules	sl	none	4	1
local	ig	index to ADS records (or granules)	sl	none	4	1
local	j	index to image pixels (j = 0, 511)	sl	none	4	1
local	j′	index to tie point pixels (j'= 0, 22)	sl	none	4	1
L2-INT-1	φ_g(ig, j)	Tie point latitude	float	deg.	4	j'= 0, 22
L2-INT-2	λ_g(ig, j)	Tie point longitude	float	deg.	4	j'= 0, 22
local	k	Index to across-track band (k = 0, 10)	sl	none	4	1
L2-INT-3	β^n(ig,k)	tie scan solar elevation, nadir	float	deg.	4	k = 0, 10
L2-INT-4	A^n(ig,k)	tie scan solar azimuth, nadir	float	deg.	4	k = 0, 10
L2-INT-5	β^f(ig,k)	tie scan solar elevation, forward	float	deg.	4	k = 0, 10
L2-INT-6	A^f(ig,k)	tie scan solar azimuth, forward	float	deg.	4	k = 0, 10
L2-INT-13	γ^n(ig,k)	tie scan satellite elevation, nadir	float	deg.	4	k = 0, 10
L2-INT-15	γ^f(ig,k)	tie scan satellite elevation, forward	float	deg.	4	k = 0, 10
local	UTCE	Scan UTC Time (byproduct not required)	char	n/a	27	1
local	DUT1E	delta UT1 for scan (byproduct not required)	char	n/a	8	1
L2-INT-25	status	status flag	sl	n/a	4	1

Table 4-2-2: Internal Data Table - Input Annotation Data Sets

4.2.3 Detailed Structure

Step 4.2.1 Read ADS records.

The required ADS records as tabulated above are read in. Angular variables are converted from the external format (fixed point in units of millidegrees for solar angles, microdegrees for lat/long) to floating point format in degrees.

For the first record of each data set the record (image scan) counter *ig* is initialised to 0. Increment by 1 for each subsequent record.

ADS #1 and #2. For each record *ig* and for each *j* = 0, 511

scn_nadir(ig, j) = [GBTR-ADS1-5](ig, j)

pxl_nadir(ig, j) = [GBTR-ADS1-6](ig, j)

scn_frwrd(ig, j) = [GBTR-ADS2-5](ig, j)

pxl_frwrd(ig, j) = [GBTR-ADS2-6](ig, j)

ADS #3: For each record *ig* and for each *j′* = 0, 22

$\varphi_g(ig, j) = 0.000001[GBTR\text{-}ADS3\text{-}5](ig, j')$

$\lambda_g(ig, j) = 0.000001[GBTR\text{-}ADS3\text{-}6](ig, j')$

ADS #5 and #6. For each record *ig* and for each *k* = 0, 10

$\beta^n(ig, k) = 0.001 \times [GBTR\text{-}ADS5\text{-}5](ig, k)$

$A^n(ig, k) = 0.001 \times [GBTR\text{-}ADS5\text{-}7](ig, k)$

$\beta^f(ig, k) = 0.001 \times [GBTR\text{-}ADS6\text{-}5](ig, k)$

$A^f(ig, k) = 0.001 \times [GBTR\text{-}ADS6\text{-}7](ig, k)$

$\gamma^n(ig, k) = 0.001 \times [GBTR\text{-}ADS5\text{-}6](ig, k)$

$\gamma^f(ig, k) = 0.001 \times [GBTR\text{-}ADS6\text{-}6](ig, k)$

(Req 4.2-1)

(The satellite azimuth is not required for level 2 processing as currently defined, but the satellite elevation is required for land surface temperature processing.)

ADS #4: For the first record of each data set the record (instrument scan) counter *sg* is initialised to 0. Increment by 1 for each subsequent record.

For each record sg copy the three (long integer) words of the UTC scan time field into the first three elements of the corresponding array, noting that the second and third elements are to be converted from ul to sl:

[MJDT[0:2]/(1:3)](*sg*) = [GBTR-ADS4-1]

(Req 4.2-2)

(Only the instrument scan times are required for Level 2 processing.)

Step 4.2.2 Convert Scan UT from Transport to Processing Format.

The CFI time conversion subroutine must now be used to convert each instrument scan time from transport to processing format. This is necessary so that the scan time can be interpolated freely within granules.

The subroutine `pl_tmjd` from the time conversion library is used to convert the scan time from transport format to processing format. The time in external (character string) format is produced as an (unwanted) byproduct.

Initialise MJDT[3]/(4) = 0.0

```
status = pl_tmjd(MJDT, MJDP, UTCE, DUT1E)
```

Check that the value of status is zero; if it is not, an input error has occurred. If it is,

```
time(sg) = MJDP[0]/(1)
                                                    (Req 4.2-3)
```

Also save the corresponding scan number from the same record:

```
scan(sg) = [GBTR-ADS4-4](sg)
                                                    (Req 4.2-4)
```

4.3 Module Definition: Assemble Regridded Brightness Temperature Arrays

4.3.1 Functional Description

This module reads in grid co-ordinates and 7 channel brightness temperatures / reflectances for forward and nadir views from the appropriate MDS of the GBTR product and arranges them in the required memory configuration.

4.3.2 Interface Definition

The module input data is read from the GBTR parameters listed in Table 4.3.1: Input Data Table - Assemble Regridded Brightness Temperature Arrays, output is internal.

Parameter ID	Variable	Name	Type	Units	Field size	Fields
GBTR-MDS1-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS1-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS1-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS1-4		image scan y coordinate	sl	m	4	1
GBTR-MDS1-5		nadir BT pixels 12 micron channel	ss	0.01K	2	512
GBTR-MDS2-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS2-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS2-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS2-4		image scan y coordinate	sl	m	4	1
GBTR-MDS2-5		nadir BT pixels 11 micron channel	ss	0.01K	2	512
GBTR-MDS3-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS3-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS3-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS3-4		image scan y coordinate	sl	m	4	1
GBTR-MDS3-5		nadir BT pixels 3.7 micron channel	ss	0.01K	2	512
GBTR-MDS4-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS4-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS4-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS4-4		image scan y coordinate	sl	m	4	1

GBTR-MDS4-5		nadir reflectance pixels 1.6 micron channel	ss	0.01%	2	512
GBTR-MDS8-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS8-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS8-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS8-4		image scan y coordinate	sl	m	4	1
GBTR-MDS8-5		forward BT pixels 12 micron channel	ss	0.01K	2	512
GBTR-MDS9-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS9-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS9-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS9-4		image scan y coordinate	sl	m	4	1
GBTR-MDS9-5		forward BT pixels 11 micron channel	ss	0.01K	2	512
GBTR-MDS10-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS10-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS10-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS10-4		image scan y coordinate	sl	m	4	1
GBTR-MDS10-5		forward BT pixels 3.7 micron channel	ss	0.01K	2	512
GBTR-MDS11-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS11-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS11-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS11-4		image scan y coordinate	sl	m	4	1
GBTR-MDS11-5		forward reflectance pixels 1.6 micron channel	ss	0.01%	2	512
GBTR-MDS15-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS15-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS15-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS15-4		image scan y coordinate	sl	m	4	1
GBTR-MDS15-5		nadir view confidence words	ss	n/a	2	512
GBTR-MDS16-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS16-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS16-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS16-4		image scan y coordinate	sl	m	4	1
GBTR-MDS16-5		forward view confidence words	ss	n/a	2	512
GBTR-MDS17-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS17-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS17-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS17-4		image scan y coordinate	sl	m	4	1
GBTR-MDS17-5		nadir view cloud/land flags	ss	n/a	2	512
GBTR-MDS18-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-MDS18-2		Record Quality Indicator	sc	n/a	1	1
GBTR-MDS18-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-MDS18-4		image scan y coordinate	sl	m	4	1
GBTR-MDS18-5		forward view cloud/land flags	ss	n/a	2	512

Table 4.3.1: Input Data Table - Assemble Regridded Brightness Temperature Arrays

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-INT-10	y(i)	image scan y co-ordinate	sl	m	4	
L2-INT-101	I(1, n; i, j)	12 µm nadir Brightness Temperature	ss array	0.01 K	2	j = 0, 511
L2-INT-102	I(2, n; i, j)	11 µm nadir Brightness Temperature	ss array	0.01 K	2	j = 0, 511
L2-INT-103	I(3, n; i, j)	3.7 µm nadir Brightness Temperature	ss array	0.01 K	2	j = 0, 511
L2-INT-104	I(4, n; i, j)	1.6 µm nadir Reflectance	ss array	0.01 %	2	j = 0, 511
L2-INT-105	I(5, n; i, j)	0.870 µm nadir Reflectance	ss array	0.01 %	2	j = 0, 511
L2-INT-106	I(6, n; i, j)	0.670 µm nadir Reflectance	ss array	0.01 %	2	j = 0, 511
L2-INT-107	I(7, n; i, j)	0.555 µm nadir Reflectance	ss array	0.01 %	2	j = 0, 511
L2-INT-111	I(1, f; i, j)	12 µm forward Brightness Temperature	ss array	0.01 K	2	j = 0, 511
L2-INT-112	I(2, f; i, j)	11 µm forward Brightness Temperature	ss array	0.01 K	2	j = 0, 511
L2-INT-113	I(3, f; i, j)	3.7 µm forward Brightness Temperature	ss array	0.01 K	2	j = 0, 511
L2-INT-114	I(4, f; i, j)	1.6 µm forward Reflectance	ss array	0.01 %	2	j = 0, 511
L2-INT-115	I(5, f; i, j)	0.870 µm forward Reflectance	ss array	0.01 %	2	j = 0, 511
L2-INT-116	I(6, f; i, j)	0.670 µm forward Reflectance	ss array	0.01 %	2	j = 0, 511
L2-INT-117	I(7, f; i, j)	0.555 µm forward Reflectance	ss array	0.01 %	2	j = 0, 511

L2-INT-171		gbtr_confidence_nadir(i, j)	ss array	flags	2	j = 0, 511
L2-INT-172		gbtr_confidence_frwrd(i, j)	ss array	flags	2	j = 0, 511
L2-INT-200		nadir_blanking_pulse(i, j)	ss array	flag	2	j = 0, 511
L2-INT-201		nadir_cosmetic(i, j)	ss array	flag	2	j = 0, 511
L2-INT-202		nadir_scan_absent(i, j))	ss array	flag	2	j = 0, 511
L2-INT-203		nadir_pixel_absent(i, j)	ss array	flag	2	j = 0, 511
L2-INT-204		nadir_packet_validation_error(i, j)	ss array	flag	2	j = 0, 511
L2-INT-205		nadir_zero_count(i, j)	ss array	flag	2	j = 0, 511
L2-INT-206		nadir_saturation(i, j)	ss array	flag	2	j = 0, 511
L2-INT-207		nadir_cal_out_of_range(i, j)	ss array	flag	2	j = 0, 511
L2-INT-208		nadir_calibration_unavailable(i, j)	ss array	flag	2	j = 0, 511
L2-INT-209		nadir_unfilled_pixel(i, j)	ss array	flag	2	j = 0, 511
L2-INT-216		frwrd_blanking_pulse(i, j)	ss array	flag	2	j = 0, 511
L2-INT-217		frwrd_cosmetic(i, j)	ss array	flag	2	j = 0, 511
L2-INT-218		frwrd_scan_absent(i, j))	ss array	flag	2	j = 0, 511
L2-INT-219		frwrd_pixel_absent(i, j)	ss array	flag	2	j = 0, 511
L2-INT-220		frwrd_packet_validation_error(i, j)	ss array	flag	2	j = 0, 511
L2-INT-221		frwrd_zero_count(i, j)	ss array	flag	2	j = 0, 511
L2-INT-222		frwrd_saturation(i, j)	ss array	flag	2	j = 0, 511
L2-INT-223		frwrd_cal_out_of_range(i, j)	ss array	flag	2	j = 0, 511
L2-INT-224		frwrd_calibration_unavailable(i, j)	ss array	flag	2	j = 0, 511
L2-INT-225		frwrd_unfilled_pixel(i, j)	ss array	flag	2	j = 0, 511
L2-INT-173		gbtr_cloud_state_nadir(i, j)	ss array	flags	2	j = 0, 511
L2-INT-174		gbtr_cloud_state_frwrd(i, j)	ss array	flags	2	j = 0, 511
L2-INT-232		nadir_land(i, j)	ss array	flag	2	j = 0, 511
L2-INT-233		nadir_cloud(i, j)	ss array	flag	2	j = 0, 511
L2-INT-234		nadir_sunglint(i, j)	ss array	flag	2	j = 0, 511
L2-INT-235		nadir_v16_histogram_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-236		nadir_v16_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-237		nadir_ir11_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-238		nadir_ir12_gross_cloud_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-239		nadir_ir11_ir12_thin_cirrus_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-240		nadir_ir37_ir12med_high_level_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-241		nadir_ir11_ir37_fog_low_stratus_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-242		nadir_ir11_ir12_view_diff_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-243		nadir_ir37_ir11_view_diff_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-244		nadir_ir11_ir12_histogram_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-248		frwrd_land(i, j)	ss array	flag	2	j = 0, 511
L2-INT-249		frwrd_cloud(i, j)	ss array	flag	2	j = 0, 511
L2-INT-250		frwrd_sunglint(i, j)	ss array	flag	2	j = 0, 511
L2-INT-251		frwrd_v16_histogram_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-252		frwrd_v16_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-253		frwrd_ir11_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-254		frwrd_ir12_gross_cloud_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-255		frwrd_ir11_ir12_thin_cirrus_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-256		frwrd_ir37_ir12med_high_level_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-257		frwrd_ir11_ir37_fog_low_stratus_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-258		frwrd_ir11_ir12_view_diff_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-259		frwrd_ir37_ir11_view_diff_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-260		frwrd_ir11_ir12_histogram_test(i, j)	ss array	flag	2	j = 0, 511
local	i	row (scan) index	integer	none	4	1
local	j	pixel index (j = 0, 511)	integer	none	4	1
L2-INT-100		nadir_fill_state(i, j)	byte	none	1	j = 0, 511
L2-INT-110		frwrd_fill_state(i, j)	byte	none	1	j = 0, 511
constant		NATURAL_PIXEL (= 0)	byte	none	1	1
constant		COSMETIC_PIXEL (= 1)	byte	none	1	1
constant		UNFILLED_PIXEL (= 2)	byte	none	1	1

Table 4.3.2: Internal Data Table - Assemble Regridded Brightness Temperature Arrays

4.3.3 Detailed Structure

4.3.3.1 Initialise Arrays

An internal data array needs to be constructed and initialised for each of the following parameters (the arrays are defined in Table 4-3-2):

- 12µ channel nadir brightness temperature
- 11µ channel nadir brightness temperature
- 3.7µ channel nadir brightness temperature
- 1.6µ channel nadir reflectance
- 0.870µ channel nadir reflectance
- 0.670µ channel nadir reflectance
- 0.555µ channel nadir reflectance
- 12µ channel forward brightness temperature
- 11µ channel forward brightness temperature
- 3.7µ channel forward brightness temperature
- 1.6µ channel forward reflectance
- 0.870µ channel forward reflectance
- 0.670µ channel forward reflectance
- 0.555µ channel forward reflectance
- nadir look confidence words
- forward look confidence words
- nadir look land / cloud flags
- forward look land / cloud flags

4.3.3.2 Input MDS records

For the first record, initialise a record counter $i = 0$.

For each subsequent record, increment i by 1.

Step 4.3.1 Read Measurement Data Sets #1 - 7.

These measurement data sets represent the nadir view channel brightness/reflectance values. For each channel *ch* = 1, 7 and for each image scan *i*:

```
FOR each pixel j = 0, 511,
       I(ch, n; i, j) = [GBTR-MDS<ch>-5](i, j)                 (Req 4.3-1)
```

No type conversion is required here.

Step 4.3.2 Read Measurement Data Sets #8 - 14.

These measurement data sets represent the forward view channel brightness/reflectance values. For each channel *ch* = 1, 7 and for each image scan *i*:

```
FOR each pixel j = 0, 511,
       [GSST-MDS<7 + ch>-5](i, j) = I(ch, f; i, j)            (Req 4.3-2)
```

No type conversion is required here.

Step 4.3.3 Read Measurement Data Sets #15, 16.

These measurement data sets represent the nadir and forward view confidence words respectively. For each image scan *i*, read in the records of confidence words:

```
FOR each pixel j = 0, 511,
       gbtr_confidence_nadir(i, j) = [GBTR-MDS15-5](i, j)
FOR each pixel j = 0, 511,
       gbtr_confidence_frwrd(i, j) = [GBTR-MDS16-5](i, j)     (Req 4.3-3)
```

Extract the individual flags from the confidence words for each pixel *j* = 0, 511. For each confidence flag, the truth value (1 = TRUE; 0 = FALSE) of the flag is to be set according to the corresponding bit of the confidence word as follows:

```
<view>_blanking_pulse(i, j) = [gbtr_confidence_<view>(i, j)](bit 0)
<view>_cosmetic(i, j) = [gbtr_confidence_<view>(i, j)](bit 1)
<view>_scan_absent(i, j) = [gbtr_confidence_<view>(i, j)](bit 2)
<view>_pixel_absent(i, j) = [gbtr_confidence_<view>(i, j)](bit 3)
<view>_packet_validation_error(i, j) =
       [gbtr_confidence_<view>(i, j)](bit 4)
<view>_zero_count(i, j) = [gbtr_confidence_<view>(i, j)](bit 5)
<view>_saturation(i, j) = [gbtr_confidence_<view>(i, j)](bit 6)
<view>_cal_out_of_range(i, j) = [gbtr_confidence_<view>(i, j)](bit 7)
<view>_calibration_unavailable(i, j) =
       [gbtr_confidence_<view>(i, j)](bit 8)
<view>_unfilled_pixel(i, j) = [gbtr_confidence_<view>(i, j)](bit 9)
```

where in each case

```
       <view> = <nadir | frwrd>                               (Req 4.3-4)
```

Set the fill state bytes for each pixel *j* = 0, 511 according to the settings of the relevant the confidence flags as follows:

```
<view>_fill_state(i, j) =
       UNFILLED_PIXEL IF <view>_unfilled_pixel(i, j) is TRUE;
<view>_fill_state(i, j) =
        COSMETIC_PIXEL IF <view>_cosmetic(i, j) is TRUE;
<view>_fill_state(i, j) = NATURAL_PIXEL otherwise,
```

where in each case

```
       <view> = <nadir | frwrd>                               (Req 4.3-5)
```

Step 4.3.4 Read Measurement Data Sets #17, 18.

These measurement data sets represent the nadir and forward view cloud/land flag words respectively. For each image scan *i*, read in the records of cloud flag words:

```
For each pixel j = 0, 511
       gbtr_cloud_state_nadir(i, j) = [GBTR-MDS17-5](i, j)
For each pixel j = 0, 511
```

```
        gbtr_cloud_state_frwrd(i, j) = [GBTR-MDS18-5](i, j)     (Req 4.3-6)
```

Extract the land and cloud cloud state flags for each pixel $j = 0, 511$:

For each cloud/land state flag, the truth value (1 = TRUE; 0 = FALSE) of the flag is to be set according to the corresponding bit of the cloud state word as follows:

```
<view>_land(i, j) = [gbtr_cloud_state_<view>(i, j)](bit 0)
<view>_cloud(i, j) = [gbtr_cloud_state_<view>(i, j)](bit 1)
<view>_sunglint(i, j) = [gbtr_cloud_state_<view>(i, j)](bit 2)
<view>_v16_histogram_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 3)
<view>_v16_spatial_coherence_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 4)
<view>_ir11_spatial_coherence_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 5)
<view>_ir12_gross_cloud_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 6)
<view>_ir11_ir12_thin_cirrus_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 7)
<view>_ir37_ir12med_high_level_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 8)
<view>_ir11_ir37_fog_low_stratus_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 9)
<view>_ir11_ir12_view_diff_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 10)
<view>_ir37_ir11_view_diff_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 11)
<view>_ir11_ir12_histogram_test(i, j) =
        [gbtr_cloud_state_<view>(i, j)](bit 12)
```

where in each case

```
        <view> = <nadir | frwrd>                                (Req 4.3-7)
```

4.4 Module Definition: Interpolate Solar Angles

4.4.1 Functional Description

The solar azimuth and elevation and satellite azimuth and elevation, all measured at the pixel, are available for a series of uniformly spaced tie point pixels in ADS #5 for the nadir view and in ADS #6 for the forward view images. The present module derives those angles that are required for level 2 processing at every scan, and at the mid-points of the bands, by linear interpolation, between these tie points. Only the solar elevation is required for most Level 2 processing, but the satellite elevation is required for LST retrieval. (If it were necessary to derive interpolated azimuths as well, account would need to be taken of the possibility of a discontinuity as the azimuth passes through 180°.)

4.4.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-AUX3-9	NGRANULE	interval between ADS records (Granule size)	sl	none	4	1

Table 4.4.1: Input Data Table - Interpolate Solar Angles

Parameter ID	Variable	Name	Type	Units	Field size	Fields
local	k	index to across-track band edge samples (k = 0, 10)	sl	none	4	1
local	k′	index to across-track band centre samples (k′= 0, 9)	sl	none	4	1
L2-INT-3	$\beta^n(ig,k)$	tie point solar elevation nadir	float	deg.	4	k = 0, 10
L2-INT-5	$\beta^f(ig,k)$	tie point solar elevation frwrd	float	deg.	4	k = 0, 10
local	ig	index to (tie) rows	sl	none	4	1
local	i	index to image rows (scans)	sl	none	4	1
L2-INT-120		nadir_band_edge_solar_elevation(i, k)	float	degrees	4	k = 0, 10
L2-INT-121		nadir_band_edge_satellite_elevation(i, k)	float	degrees	4	k = 0, 10
L2-INT-140		frwrd_band_edge_solar_elevation(i, k)	float	degrees	4	k = 0, 10
L2-INT-141		frwrd_band_edge_satellite_elevation(i, k)	float	degrees	4	k = 0, 10
L2-INT-124		nadir_band_centre_solar_elevation(i, k′)	float	degrees	4	k′ = 0, 9
L2-INT-125		nadir_band_centre_satellite_elevation(i, k′)	float	degrees	4	k′ = 0, 9
L2-INT-144		frwrd_band_centre_solar_elevation(i, k′)	float	degrees	4	k′ = 0, 9
L2-INT-145		frwrd_band_centre_satellite_elevation(i, k′)	float	degrees	4	k′ = 0, 9

Table 4.4.2: Internal Data Table - Interpolate Solar Angles

4.4.3 Detailed Structure

Intermediate values are now calculated by linear interpolation. The nadir and forward scans are treated separately. Level 1b processing will have ensured that each ADS record corresponds to a single image row with

$$i = ig \cdot NGRANULE$$

where *i* indexes the image scans, and *ig* is an index to the ADS records, both counting from 0.

The elevation values read in from the ADS records may contain exception values (-999). If a solar elevation of -999.0 appears at either end of an interpolation interval, the interpolated values in the interval should all be set to -999.0. This ensures that the interpolated values are consistent with the values derived at Level 1B in the presence of exception values.

(Implementation note: this may be achieved by initialising the arrays *<view>_band_edge_solar_elevation* and *<view>_band_centre_solar_elevation to* -999.0)

Step 4.4.1. Interpolate band edge solar angles.

The solar elevations for the intermediate rows can be determined by linear interpolation as follows:

for each k = 0, 1, ..., 10 and for j = 0, 1, ... $NGRANULE$ - 1

calculate

$$w = j \,/\, NGRANULE$$

$$i = ig \cdot NGRANULE + j$$

If [L2-INT-3](ig, k) equals -999.0 or [L2-INT-3](ig + 1, k) equals -999.0 (these are the nadir view tie point solar elevations at the two ends of the interpolation interval) then set *nadir_band_edge_solar_elevation(i, k)* = -999.0

otherwise

$$nadir\_band\_edge\_solar\_elevation(i,k) = \beta^n(ig,k) + w \cdot \left\{\beta^n(ig+1,k) - \beta^n(ig,k)\right\}$$

If [L2-INT-5](*ig*, *k*) equals -999.0 or [L2-INT-5](*ig* + 1, *k*) equals -999.0 (these are the forward view tie point solar elevations at the two ends of the interpolation interval) then set

frwrd_band_edge_solar_elevation(i, k) = -999.0

otherwise

$$frwrd\_band\_edge\_solar\_elevation(i,k) = \beta^f(ig,k) + w \cdot \left\{\beta^f(ig+1,k) - \beta^f(ig,k)\right\}$$

(Req 4.4-1)

Step 4.4.2. Interpolate band centre solar angles.

The band centre values are then given as follows.

If *<view>_band_edge_solar_elevation(i, k′)* equals -999.0 or *<view>_band_edge_solar_elevation(i, k′ + 1)* equals -999.0 then

<view>_band_centre_solar_elevation(i, k′) = -999.0

otherwise

<view>_band_centre_solar_elevation(*i*, *k′*) =
0.5{*<view>_band_edge_solar_elevation*(*i*, *k′*) +
<view>_band_edge_solar_elevation(*i*, *k′+1*)}

for *k′* = 0, 9.

(Req 4.4-2)

Step 4.4.3. Interpolate band edge satellite angles.

The satellite elevations for the intermediate rows are determined by linear interpolation in the same way as the solar elevations, but with [L2-INT-13] in place of [L2-INT-3] and with [L2-INT-15] in place of [L2-INT-5].

for each *k* = 0, 1, ..., 10 and for *j* = 0, 1, ... *NGRANULE* - 1

calculate

$$w = j \,/\, NGRANULE$$

$$i = ig \cdot NGRANULE + j$$

If [L2-INT-13](*ig*, *k*) equals -999.0 or [L2-INT-13](*ig* + 1, *k*) equals -999.0 then set

nadir_band_edge_satellite_elevation(i, k) = -999.0

otherwise

$$nadir\_band\_edge\_satellite\_elevation(i,k) = \gamma^n(ig,k) + w \cdot \left\{\gamma^n(ig+1,k) - \gamma^n(ig,k)\right\}$$

If [L2-INT-15](ig, k) equals -999.0 or [L2-INT-15](ig + 1, k) equals -999.0 then set

frwrd_band_edge_satellite_elevation(i, k) = -999.0

otherwise

$$frwrd\_band\_edge\_satellite\_elevation(i,k) = \gamma^f(ig,k) + w \cdot \left\{\gamma^f(ig+1,k) - \gamma^f(ig,k)\right\}$$

(Req 4.4-3)

Step 4.4.4. Interpolate band centre satellite angles.

The band centre values are then given as follows.

If *<view>_band_edge_satellite_elevation(i, k′)* equals -999.0 or *<view>_band_edge_satellite_elevation(i, k′ + 1)* equals -999.0 then

<view>_band_centre_satellite_elevation(i, k′) = -999.0

otherwise

<view>_band_centre_satellite_elevation(*i*, *k′*) = 0.5{*<view>_band_edge_satellite_elevation*(*i*, *k′*) + *<view>_band_edge_satellite_elevation*(*i*, *k′*+*1*)}

for k'= 0, 9.

(Req 4.4-4)

4.5 Module Definition: Interpolate Image Pixel Position

4.5.1 Functional Description

The (geodetic) latitudes and longitudes of a series of uniformly space tie point pixels are available in ADS #3. This module derives the latitude and longitude of each of image pixel by linear interpolation, in two dimensions, between these tie points. In the case of longitude account must be taken of the possibility that the 180 degree meridian intersects the image scan.

4.5.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
GBTR-ADS3-1		UTC nadir time 1	sl	days	4	1
GBTR-ADS3-2		UTC nadir time 2	ul	s	4	1
GBTR-ADS3-3		UTC nadir time 3	ul	micros	4	1
GBTR-ADS3-4		image scan y coordinate	sl	km	4	1
GBTR-ADS3-5		tie point latitudes	sl	mdeg	4	23
GBTR-ADS3-6		tie point longitudes	sl	mdeg	4	23
GBTR-ADS3-7		latitude corrections	ss	mdeg	2	23
GBTR-ADS3-8		longitude corrections	ss	mdeg	2	23
GBTR-ADS3-9		Topographic Altitude	ss	metres	2	23
L2-AUX3-9	NGRANULE	interval between ADS records (Granule size)	sl	none	4	1

Table4.5-1: Input Data Table - Interpolate Image Pixel Position

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-INT-1	$\varphi_g(i_g, j_g)$	Tie point latitude	float	degrees	4	j_g = 0, 22
L2-INT-2	$\lambda_g(i_g, j_g)$	Tie point longitude	float	degrees	4	j_g = 0, 22
local	i_g	index to tie scans	sl	none	4	1
local	j_g	index to tie point pixels (jg = 0, 22)	sl	none	4	1
local	i	index to image scans	sl	none	4	1
local	j	index to image pixels (j = 0, 511)	sl	none	4	1
L2-INT-160	$\varphi(i, j)$	image latitude	float	degrees	4	j = 0, 511
L2-INT-161	$\lambda(i, j)$	image longitude	float	degrees	4	j = 0, 511
local	w_x	interpolation weight in x	float	none	4	1
local	w_y	interpolation weight in y	float	none	4	1
local	φ_1	Intermediate latitude	float	degrees	4	1
local	φ_2	Intermediate latitude	float	degrees	4	1
local	λ_1	Intermediate longitude	float	degrees	4	1
local	λ_2	Intermediate longitude	float	degrees	4	1

Table4.5-2: Internal Data Table - Interpolate Image Pixel Position

4.5.3 Detailed Structure

Step 4.5.1. Interpolate pixel latitudes.

Given an image scan and pixel number *i*, *j* define

i_g = integer part of *i*/*NGRANULE*

$$w_y = (i / NGRANULE) - i_g$$

j_g = integer part of (*j* + 19)/25

$$w_x = (j + 19) / 25 - j_g$$

Interpolate the geocentric latitudes as follows:

$$\varphi(i, j) = \varphi_1 + w_y\{\varphi_2 - \varphi_1\}$$

where

$$\varphi_1 = \varphi_g(i_g, j_g) + w_x\left\{\varphi_g(i_g, j_g + 1) - \varphi_g(i_g, j_g)\right\}$$

and

$$\varphi_2 = \varphi_g(i_g + 1, j_g) + w_x\left\{\varphi_g(i_g + 1, j_g + 1) - \varphi_g(i_g + 1, j_g)\right\}$$

(Req 4.5-1)

Step 4.5.2. Interpolate pixel longitudes

Longitude is treated similarly unless the meridian is present:

$$\lambda(i, j) = \lambda_1 + w_y\{\lambda_2 - \lambda_1\}$$

where

$$\lambda_1 = \lambda_g(i_g, j_g) + w_x\left\{\lambda_g(i_g, j_g + 1) - \lambda_g(i_g, j_g)\right\}$$

and

$$\lambda_2 = \lambda_g(i_g + 1, j_g) + w_x \left\{ \lambda_g(i_g + 1, j_g + 1) - \lambda_g(i_g + 1, j_g) \right\}$$

A test for the presence of the meridian is that

$$(\lambda_{max} - \lambda_{min}) > 180.0$$

where λ_{max} and λ_{min} are respectively the greatest and least of

$$\lambda_g(i_g, 0),\ \lambda_g(i_g, 22),\ \lambda_g(i_g + 1, 0),\ \lambda_g(i_g + 1, 22).$$

(Req 4.5-2)

In this case 360.0 is added to each of the grid longitudes that is initially negative before it is substituted in the above equations. The resultant interpolated longitude is translated into the range -180.0 to 180.0 degrees by subtracting 360.0 if its value exceeds 180.0.

4.6 Module Definition: Gridded SST / Vegetation Retrieval

4.6.1 Functional Description

To derive the sea surface temperature (SST) or, over land, the land surface temperature (LST) and vegetation index (NDVI), at 1 km resolution, using cloud free data.

The derivation of SSTs uses the 11 and 12 micron channels for day time data and for night time data the 11, 12 and 3.7 micron channels. For each 1 km resolution element two results are obtained, one using the combined nadir and forward views and the other using the nadir view alone.

The SSTs are calculated using preset retrieval coefficients. These coefficients are provided for both nadir only and combined view SSTs and are a function of latitude and viewing angle.

Smoothing is applied by smoothing the difference between the calculated SST and the 11 micron brightness temperature. The effect is to smooth the atmospheric correction.

The LST is calculated using an algorithm developed by Prata and described in Reference RD3. This algorithm is similar to that used for the nadir only SST retrieval, in that it uses the 11 and 12 micron nadir view brightness temperatures in conjunction with pre-defined retrieval coefficients. However, the selection of the retrieval coefficients is more complicated than for the SST retrieval. The coefficients depend on a surface classification and on a seasonal vegetation index, both of which are defined in the form of maps at 0.5 degree resolution, and they also depend weakly on the precipitable water vapour content derived from climatological data. The surface class, vegetation index and precipitable water vapour are supplied as auxiliary data sets in the LST Retrieval Coefficient Data Product. Different coefficients may be supplied for day and night retrievals. Note that smoothing is not applied to the LST values in the present algorithm.

The NVDIs are calculated using the nadir .67 and .87 micron channels and the results are returned in the combined image land pixels.

In cloudy conditions the cloud top temperature is returned in the nadir image field and the cloud top height in the combined view image field.

4.6.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-AUX1-1	a[i, j, 0]	sst_retrieval_a[3, 38, 0]	float	K/100	4	114
L2-AUX1-2	a[i, j, 1]	sst_retrieval_a[3, 38, 1]	float	none	4	114
L2-AUX1-3	a[i, j, 2]	sst_retrieval_a[3, 38, 2]	float	none	4	114
L2-AUX1-4	b[i, j, 0]	sst_retrieval_b[3, 38, 0]	float	k/100	4	114
L2-AUX1-5	b[i, j, 1]	sst_retrieval_b[3, 38, 1]	float	none	4	114
L2-AUX1-6	b[i, j, 2]	sst_retrieval_b[3, 38, 2]	float	none	4	114
L2-AUX1-7	b[i, j, 3]	sst_retrieval_b[3, 38, 3]	float	none	4	114
L2-AUX1-8	c[i, j, 0]	sst_retrieval_c[3, 38, 0]	float	K/100	4	114
L2-AUX1-9	c[i, j, 1]	sst_retrieval_c[3, 38, 1]	float	none	4	114
L2-AUX1-10	c[i, j, 2]	sst_retrieval_c[3, 38, 2]	float	none	4	114
L2-AUX1-11	c[i, j, 3]	sst_retrieval_c[3, 38, 3]	float	none	4	114
L2-AUX1-12	c[i, j, 4]	sst_retrieval_c[3, 38, 4]	float	none	4	114
L2-AUX1-13	d[i, j, 0]	sst_retrieval_d[3, 38, 0]	float	K/100	4	114
L2-AUX1-14	d[i, j, 1]	sst_retrieval_d[3, 38, 1]	float	none	4	114
L2-AUX1-15	d[i, j, 2]	sst_retrieval_d[3, 38, 2]	float	none	4	114
L2-AUX1-16	d[i, j, 3]	sst_retrieval_d[3, 38, 3]	float	none	4	114
L2-AUX1-17	d[i, j, 4]	sst_retrieval_d[3, 38, 4]	float	none	4	114
L2-AUX1-18	d[i, j, 5]	sst_retrieval_d[3, 38, 5]	float	none	4	114
L2-AUX1-19	d[i, j, 6]	sst_retrieval_d[3, 38, 6]	float	none	4	114
L2-AUX6-1	j	pixel index (j)	ss	none	2	512
L2-AUX6-2	map(j)	Across-track band index (map index)	ss	none	2	512
L2-AUX3-11		TROPICAL_INDEX (Initialise to 12.5)	float	deg.	4	1
L2-AUX3-12		TEMPERATE_INDEX (Initialise to 37)	float	deg.	4	1
L2-AUX3-13		POLAR_INDEX (Initialise to 70)	float	deg.	4	1
L2-AUX3-17	Smooth_fac	Smoothing scaling factor	ss	none	2	1
	The following parameters are required by the Land Surface Temperature algorirthm.					
L2-AUX10-1	d	Water vapour factor for LST retrieval	float	none	4	1
L2-AUX10-2	m	Angle factor for LST retrieval	ss	none	2	1
L2-AUX10-3	N_CLASS	Number of vegetation classes for LST	ss	none	2	1
L2-AUX5-1		Coefficient A0 (day-time) for LST	float	K	4	1
L2-AUX5-2		Coefficient A1 (day-time) for LST	float	none	4	1
L2-AUX5-3		Coefficient A2 (day-time) for LST	float	none	4	1
L2-AUX5-4		Coefficient A0 (night-time) for LST	float	K	4	1
L2-AUX5-5		Coefficient A1 (night-time) for LST	float	none	4	1
L2-AUX5-6		Coefficient A2 (night-time) for LST	float	none	4	1
L2-AUX6-1		Vegetation class index [360][720] for LST	ss	n/a	2	720
L2-AUX7-1		Vegetation fraction[12][360][720]	ss	0.001	2	720
L2-AUX8-1		Precipitable water[12][360][720]	ss	0.01 mm	2	720
L2-AUX9-1		Topographic Variance Flag[360][720]	ss	n/a	2	720

Table 4-6-1: Input Data including SST Retrieval Coefficients.

Parameter ID	Variable	Name	Type	Units	Field size	Fields
local	a(i, j, q)	averaged sst retrieval a coefficients	float	mixed	4	342
local	b(i, j, q)	averaged sst retrieval b coefficients	float	mixed	4	456
local	c(i, j, q)	averaged sst retrieval c coefficients	float	mixed	4	570
local	d(i, j, q)	averaged sst retrieval d coefficients	float	mixed	4	798
L2-INT-101	I(ir12, n; i, j)	nadir ir12 Brightness Temp.	ss	0.01 K	2	j = 0, 511
L2-INT-102	I(ir11, n; i, j)	nadir ir11 Brightness Temp.	ss	0.01 K	2	j = 0, 511
L2-INT-103	I(ir37, n; i, j)	nadir ir37 Brightness Temp.	ss	0.01 K	2	j = 0, 511
L2-INT-111	I(ir12, f; i, j)	forward ir12 Brightness Temp.	ss	0.01 K	2	j = 0, 511

L2-INT-112	I(ir11, f; i, j)	forward ir11 Brightness Temp.	ss	0.01 K	2	j = 0, 511
L2-INT-113	I(ir37, f; i, j)	forward ir37 Brightness Temp.	ss	0.01 K	2	j = 0, 511
L2-INT-105	I(v870, n; i, j)	regridded nadir v870 Reflectance	ss	%/100	2	j = 0, 511
L2-INT-106	I(v670, n; i, j)	regridded nadir v670 Reflectance	ss	%/100	2	j = 0, 511
L2-INT-115	I(v870, f; i, j)	regridded forward v870 Reflectance	ss	%/100	2	j = 0, 511
L2-INT-116	I(v670, f; i, j)	regridded forward v670 Reflectance	ss	%/100	2	j = 0, 511
L2-INT-100		nadir_fill_state(i, j)	byte	n/a	1	j = 0, 511
L2-INT-110		frwd_fill_state(i, j)	byte	n/a	1	j = 0, 511
L2-INT-160	φ(i, j)	image_latitude(i, j)	float	degrees	4	j = 0, 511
L2-INT-121		nadir_band_edge_satellite_elevation(i, k)	float	degrees	4	k = 0, 10
L2-INT-124		nadir_band_centre_solar_elevation(i, k)	float	degrees	4	k = 0, 9
L2-INT-141		frwd_band_edge_satellite_elevation(i, k)	float	degrees	4	k = 0, 10
L2-INT-144		frwd_band_centre_solar_elevation(i, k)	float	degrees	4	k = 0, 9
		nadir cloud state flags:				
L2-INT-232	nadir_land(i, j)	nadir land flag	ss	n/a	2	j = 0, 511
L2-INT-233	nadir_cloud(i, j)	nadir cloud flag	ss	n/a	2	j = 0, 511
L2-INT-235		nadir_v16_histogram_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-236		nadir_v16_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-242		nadir_ir11_ir12_view_diff_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-244		nadir_ir11_ir12_histogram_test(i, j)	ss array	flag	2	j = 0, 511
		forward cloud state flags				
L2-INT-248	frwd_land(i, j)	frwd land flag	ss	n/a	2	j = 0, 511
L2-INT-249	frwd_cloud(i, j)	frwd cloud flag	ss	n/a	2	j = 0, 511
L2-INT-251		frwd_v16_histogram_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-252		frwd_v16_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-258		frwd_ir11_ir12_view_diff_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-260		frwd_ir11_ir12_histogram_test(i, j)	ss array	flag	2	j = 0, 511
L2-INT-60	band(j)	number of across track band (or strip)	sl	none	4	j = 0, 511
L2-INT-61	map(j)	across-track mapping	sl	none	4	j = 0, 512
local	i	index to image scans	sl	none	4	1
local	j	index to image pixels, j = 0, 1, ...511	sl	none	4	1
local	k	index to across-track bands	sl	none	4	1
local	zone	latitude zone index, zone = 0, 1, 2	sl	none	4	1
local	T0	tropical_sst	float	deg.	4	1
local	T1	temperate_sst	float	deg.	4	1
local	T2	polar_sst	float	deg.	4	1
local	w	interpolation weight	float	none	4	1
local		smoothed_gsst_image(i, j)	ss	0.01 K	2	j = 0, 511
L2-INT-270		nadir_image_field(i, j)	ss	0.01 K	2	j = 0, 511
L2-INT-271		combined_image_field(i, j)	ss	mixed	2	j = 0, 511
		GSST confidence flags:				
L2-INT-280		nadir_image_valid(i, j)	flag	n/a	2	j = 0, 511
L2-INT-281		nadir_only_sst_uses_ir37(i, j)	flag	n/a	2	j = 0, 511
L2-INT-282		combined_image_valid(i, j)	flag	n/a	2	j = 0, 511
L2-INT-283		combined_view_sst_uses_ir37(i, j)	flag	n/a	2	j = 0, 511
L2-INT-284		land(i, j)	flag	n/a	2	j = 0, 511
L2-INT-285		nadir_view_cloudy(i, j)	flag	n/a	2	j = 0, 511
L2-INT-286		nadir_view_blanking_pulse(i, j)	flag	n/a	2	j = 0, 511
L2-INT-287		nadir_view_cosmetic(i, j)	flag	n/a	2	j = 0, 511
L2-INT-288		frwd_view_cloudy(i, j)	flag	n/a	2	j = 0, 511
L2-INT-289		frwd_view_blanking_pulse(i, j)	flag	n/a	2	j = 0, 511
L2-INT-290		frwd_view_cosmetic(i, j)	flag	n/a	2	j = 0, 511
L2-INT-291		gsst_v16_cloud_test(i, j)	flag	n/a	2	j = 0, 511
L2-INT-292		gsst_nadir_frwd_cloud_test(i, j)	flag	n/a	2	j = 0, 511
L2-INT-293		gsst_ir11_histogtram_test(i, j)	flag	n/a	2	j = 0, 511
L2-INT-294		topographic_variance(i, j)	flag	n/a	2	j = 0, 511
L2-INT-295		extended_land(i, j)	flag	n/a	2	j = 0, 511
L2-INT-26	time(sg)	scan UTC	double	days	8	per sg
L2-INT-161	λ(i, j)	image_longitude(i, j)	float	deg.	4	j = 0, 511

L2-INT-470		vegetation_class(lat_index, lon_index)	ss	none	2	360 × 720
L2-INT-471		vegetation_fraction(lat_index, lon_index)	ss	0.001	4	360 × 720
L2-INT-472		precipitable_water(lat_index, lon_index)	ss	0.01 mm	2	360 × 720
L2-INT-473		Topographic_flag(lat_index, lon_index)	ss	none	2	360 × 720
L2-INT-474	lat_index	Latitude index: lat_index = 0, …359	sl	none	4	1
L2-INT-475	lon_index	Longitude index: lon_index = 0, 719	sl	none	4	1
L2-INT-489	disp_lat_index	Displaced latitude index: = 0, …359	sl	none	4	1
L2-INT-490	disp_lon_index	Displaced longitude index: = 0, 719	sl	none	4	1
L2-INT-476	month	Index to month: month = 0, 11	sl	none	4	1
L2-INT-477	sun_elev	Solar elevation at land pixel	float	degrees	4	1
L2-INT-478	sat_elev	Satellite elevation at land pixel	float	degrees	4	1
L2-INT-479	night	Day/night flag	sl	none	2	1
L2-INT-480	n	Non-linear exponent	float	none	4	1
L2-INT-481	f	Vegetation fraction at pixel	float	none	4	1
L2-INT-482	pw	Precipitable water at pixel	float	cm	4	1
L2-INT-482	coeff(class, i, j, 0)	Sub-array of coefficients A0	float	0.01K	4	64
L2-INT-484	coeff(class, i, j, 1)	Sub-array of coefficients A1	float	none	4	64
L2-INT-485	coeff(class, i, j. 2)	Sub-array of coefficients A2	float	none	4	64
L2-INT-486	w	Interpolation weight	float	none	4	1
L2-INT-487	a(k)	Retrieval coefficients for pixel	float	mixed	4	1
L2-INT-488	lst	Land surface temperature at pixel	float	0.01K	4	1
Local	pw00	Precipitable water sample value	ss	0.01 mm	2	1
Local	pw01	Precipitable water sample value	ss	0.01 mm	2	1
Local	pw10	Precipitable water sample value	ss	0.01 mm	2	1
Local	pw11	Precipitable water sample value	ss	0.01 mm	2	1
Local	q	Latitude argument for bilinear interpolation	float	none	4	1
Local	p	Longitude argument for bilinear interpolation	float	none	4	1
Local	class	Index to table of cefficients	sl	none	4	1

Table 4.6.2: Internal Data Table - Gridded SST/Vegetation Retrieval

4.6.3 Detailed Structure

Step 4.6.1 Read in the retrieval coefficients.

Step 4.6.1.1 Read in the SST retrieval coefficients.

This is done once at initialisation. Retrieval coefficients are specified for tropical, temperate and polar latitudes and for 38 bands or strips running parallel to the ground track, on each side, corresponding to different viewing angles. Also different sets are needed for day/night and for nadir only/combined view retrievals, as follows.

Latitude zone

0 tropical
1 temperate
2 polar

Set

a nadir only; day
b nadir only; night
c combined view; day
d combined view; night

First the mapping array must be read in. Open the data set L2-AUX6 and set

map(j) = [L2-AUX6-2](j), j = 0, 511

(Note that [L2-AUX6-1](j) = j, j = 0, 511)

Open the file of retrieval coefficients L2-AUX1.

The 'a' coefficient set are read in as follows.
zone = 0, 1, 2 (outer loop)
Across-track band *k* = 0 to 37 (inner loop)
a(zone, k, 0) = [L2-AUX1-1](*zone, k*)
a(zone, k, 1) = [L2-AUX1-2](*zone, k*)
a(zone, k, 2) = [L2-AUX1-3](*zone, k*)

Similarly read in the b, c and d sets of coefficients.
b(zone, k, 0:3) = L2-AUX1-<4 - 7>](*zone, k*)
c(zone, k, 0:4) = L2-AUX1-<8 - 12>](*zone, k*)
d(zone, k, 0:6) = L2-AUX1-<13 - 19>](*zone, k*)

Step 4.6.1.2 Read in coefficients and auxiliary tables for LST retrieval.

Step 4.6.1.2.1 Read in coefficients.

For each of the *N_CLASS* vegetation classes there are two records, for vegetation and for bare soil. Open the file of retrieval coefficients L2-AUX5.

The LST coefficient set is read in as follows.

for *class* = 0, *N_CLASS* - 1 (outer loop)
for *i* = 0, 1 (inner loop)
coeff(class, i, 0, 0) = [L2-AUX5-1]
coeff(class, i, 0, 1) = [L2-AUX5-2]
coeff(class, i, 0, 2) = [L2-AUX5-3]
coeff(class, i, 1, 0) = [L2-AUX5-4]
coeff(class, i, 1, 1) = [L2-AUX5-5]
coeff(class, i, 1, 2) = [L2-AUX5-6]

(Note: *i* = 0 are vegetation coefficients; *i* = 1 are coefficients for bare soil.)

Step 4.6.1.2.2 Determine month index.

Using a suitable calendar function, determine the month (*month* = 0, … 11) in which the data were collected from the scan time of start of data *time*(0)=[L2-INT-26](0):

month = *month*(*time*(0))

Step 4.6.1.2.3 Read in auxiliary files.

Note that in the cases of data sets L2-AUX7 and L2-AUX8 only one plane of data, that corresponding to the current month, is required in memory for a given run of the processor.

Read in Vegetation Class Index: Open the vegetation class file L2-AUX6.

for each latitude index *i* = 0, 359
vegetation_class(*i, j*) = [L2-AUX6-1](*j*) for all *j* of record *i*.

Read in Vegetation Fraction Table: Open the file of vegetation fraction data L2-AUX7.

for each latitude index $i = 0, 359$
 select record ($360 \times month + i$)
 vegetation_fraction(i, j) = [L2-AUX7-1](j) for all j of selected record.

Read in Precipitable Water Data: Open the file of precipitable water data L2-AUX8.

for each latitude index $i = 0, 359$
 select record ($360 \times month + i$)
 precipitable_water(i, j) = [L2-AUX8-1](j) for all j of selected record.

Read in Topographic Variance Flag: Open the file of topographic variance flags L2-AUX9.

for each latitude index $i = 0, 359$
 topographic_flag(i, j) = [L2-AUX9-1](j) for all j of record i.

Step 4.6.2 Calculate across-track band number.

The across-track band number for each pixel in the regridded image is given by

```
band(j)  = 0 IF j < 6
band(j)  = integer part of (j - 6 ) / 50   IF  6 ≤ j < 506
band(j)  = 9 IF j ≥ 506
```

where j is the pixel across track index.

(Req 4.6-1)

Step 4.6.3 Clear the gsst confidence flags.

The GSST confidence flags L2-INT-280 and L2-INT-282 are set to indicate data invalid for each pixel in the GSST arrays.

```
nadir_image_valid(i, j) = FALSE
combined_image_valid(i, j) = FALSE                          (Req 4.6-2)
```

(Implementation note: it may be convenient to initialise all the GSST confidence flags L2-INT-280 to L2-INT-293 inclusive at this point, although it is not logically necessary in terms of the way the algorithms are formulated in the following.)

Step 4.6.4 Derive nadir only sea-surface temperature image.

Each pixel (i, j) in the nadir image field is processed as follows.

The latitude and longitude indices are extracted and used to identify the surface classification. The objective here is to identify those pixels that are flagged as sea pixels but represent inland lakes. The extended land flag is set if the pixel is either a land pixel or an inland lake: extended_land = (land OR {(NOT land) AND class = 14})

If *nadir_land*(i, j) = FALSE then

lat_index = integer part of [(*image_latitude*(i, j) + 90.0) × 2.0]

lon_index = integer part of [(*image_longitude*(i, j) + 180.0) × 2.0]

class = *vegetation_class*(*lat_index*, *lon_index*)

extended_land(i, j) = (*class* = 14)

else

extended_land(i, j) = *nadir_land*(i, j)

IF over cloud and the extended land flag is not set, and the 11 micron brightness temperature is valid then the 11 micron brightness temperature is used as an estimate of cloud top temperature ie.

```
if nadir_cloud(i, j) AND NOT extended_land(i, j) = TRUE
AND I(ir11, n; i, j) > 0
then
nadir_image_field(i, j) = I(ir11, n; i, j)
nadir_image_valid(i, j) = TRUE                          (Req 4.6-3)
```

IF the extended land flag is set then execute Step 4.6.9 below to derive the land surface temperature:

```
IF  extended_land(i, j) =  TRUE
THEN execute Step 4.6.9
                                                        (Req 4.6-4)
```

ELSE the pixel is over open sea and NOT cloudy. Determine the nadir-only sea surface temperature SST as follows and assign it to nadir_image_field(i, j).

Step 4.6.4.1

IF the absolute value of the pixel latitude φ(i, j) is less than TROPICAL_INDEX retrieve the tropical sea surface temperature (using step 4.6.4.5 below).

Step 4.6.4.2

ELSE IF the absolute value of the pixel latitude is less than TEMPERATE_INDEX but is not less than TROPICAL_INDEX, two retrievals are made using the retrieval coefficients for both the tropical and mid-latitude zones. If these two retrievals are tropical_sst and temperate_sst respectively, the final value for the retrieved sst is given by linear interpolation between them, as follows:

$$w = \frac{(abs(\varphi(i,j)) - TROPICAL\_INDEX)}{(TEMPERATE\_INDEX - TROPICAL\_INDEX)}$$

$$SST = tropical\_sst + w \times (temperate\_sst - tropical\_sst)$$

(Req 4.6-5)

Step 4.6.4.3

ELSE IF the absolute value of the pixel latitude is less than POLAR_INDEX but not less than TEMPERATE_INDEX, two retrievals are made using the retrieval coefficients for the high-latitude and mid-latitude regions. If these two retrievals are polar_sst and temperate_sst respectively, the final value for the retrieved sst is given by

$$w = \frac{(abs(\varphi(i,j)) - TEMPERATE\_INDEX)}{(POLAR\_INDEX - TEMPERATE\_INDEX)}$$

$$SST = temperate\_sst + w \times (polar\_sst - temperate\_sst)$$

(Req 4.6-6)

Step 4.6.4.4

ELSE retrieve the polar sea surface temperature using step 4.6.4.5 below.

Step 4.6.4.5. Retrieve nadir only SST

This step is executed whenever an SST retrieval is called for in Steps 4.6.4.1 to 4.6.4.4.

Retrievals use the 11, 12 and 3.7 micron brightness temperatures.

The latitude zone, 0 for tropical, 1 for temperate, 2 for polar.

The across-track band mapping index k = map (j), k = 0 to 37.

The nadir view solar elevation angle.

Negative data values indicate invalid data. If either the 11 or 12 micron brightness temperatures are invalid then the calculation is abandoned and the nadir_image_valid flag remains FALSE. Otherwise the calculation can proceed as follows.

Check the nadir solar elevation angle and if the pixel is in night-time (nadir band centre solar elevation(i, k) is < 0.) and the 3.7 micron brightness temperature is valid use the 3.7 micron channel with the b coefficient set.

$$
\begin{aligned}
sst \;=\; & b(zone, k, 0) * 100. + \\
& b(zone, k, 1) * I(ir11, n, i, j) + \\
& b(zone, k, 2) * I(ir12, n, i, j) + \\
& b(zone, k, 3) * I(ir37, n, i, j)
\end{aligned}
$$

```
nadir_only_sst_uses_ir37(i, j )  =  TRUE
nadir_image_valid(i, j )  =  TRUE.
                                                  (Req 4.6-7)
```

If NOT using 3.7 micron channel THEN a 2-channel retrieval is performed and the a coefficient set is required.

$$
\begin{aligned}
sst \;=\; & a(zone, k, 0) * 100. + \\
& a(zone, k, 1) * I(ir11, n, i, j) + \\
& a(zone, k, 2) * I(ir12, n, i, j)
\end{aligned}
$$

```
nadir_only_sst_uses_ir37(i, j )  =  FALSE
nadir_image_valid(i, j )  =  TRUE.                (Req 4.6-8)
```

Step 4.6.5 Derive combined view sea-surface temperature image

Each pixel (i, j) in the image field is processed as follows.

IF the nadir cloud flag [L2-INT-233] for the pixel is set (nadir_cloud(i, j) = TRUE) and the extended land flag *extended_land*(*i*, *j*)is NOT set THEN set

```
combined_image_field(i, j) = 0
combined_image_valid(i, j) = FALSE                (Req 4.6-8.1)
```

(Note: for cloudy pixels, the combined image field is reserved for the cloud top height. However, the algorithm for determining cloud top height is not yet defined, and so the combined image field is currently set to to zero in this case. The eventual algorithm is expected to make use of a look-up table of temperature versus height.)

IF over land (*extended_land*(*i*, *j*) = TRUE) THEN calculate the NDVI for the pixel (step 4.6.6).

Otherwise the pixel is not cloudy and over sea. Determine the combined view SST as follows and assign it to combined_image_field(i, j).

Step 4.6.5.1

IF the absolute value of the pixel latitude is LESS THAN TROPICAL_INDEX retrieve the combined view tropical sea surface temperature. Combined view temperature is derived according to step 4.6.5.5 below.

Step 4.6.5.2

ELSE if the absolute value of the pixel latitude is LESS THAN TEMPERATE_INDEX but is NOT LESS THAN TROPICAL_INDEX, two retrievals are made using the retrieval coefficients for both the tropical and mid-latitude zones. IF these two retrievals are tropical_sst and temperate_sst respectively, the final value for the retrieved sst is given by linear interpolation as follows:

$$w = \frac{(abs(\varphi(i,j)) - TROPICAL\_INDEX)}{(TEMPERATE\_INDEX - TROPICAL\_INDEX)}$$

$$sst = tropical\_sst + w \times (temperate\_sst - tropical\_sst)$$

(Req 4.6-9)

Step 4.6.5.3

ELSE IF the absolute value of the latitude [L2-INT-160] is less than POLAR_INDEX but not less than TEMPERATE_INDEX, two retrievals are made using the retrieval coefficients for the high-latitude and mid-latitude regions. If these two retrievals are polar_sst and temperate_sst respectively, the final value for the retrieved sst is given by

$$w = \frac{(abs(\varphi(i,j)) - TEMPERATE\_INDEX)}{(POLAR\_INDEX - TEMPERATE\_INDEX)}$$

$$sst = temperate\_sst + w \times (polar\_sst - temperate\_sst)$$

(Req 4.6-10)

Step 4.6.5.4

ELSE retrieve the polar combined view sea-surface temperature using step 4.6.5.5 below.

Step 4.6.5.5 Combined view retrieval

This step is executed whenever an SST retrieval is called for in Steps 4.6.5.1 to 4.6.5.4.

Combined view retrievals use the nadir and forward 11, 12 and 3.7 micron brightness temperatures; the latitude zone = 0 for tropical, 1 for temperate, 2 for polar; the across track band mapping index number k = 0 to 37; and the nadir and forward view solar elevation angles.

If the 11 or 12 micron brightness temperatures in either the nadir or forward view are invalid then the calculation is abandoned and the combined_image_valid flag remains FALSE. Otherwise the calculation can proceed as follows.

Use the 3.7 micron channel if the data is valid and the pixel is in night-time in both nadir and forward views, ie

I(ir37, n; i, j) > 0 AND I(ir37, f; i, j) > 0 AND
nadir_band_centre_solar_elevation(i, k) < 0 AND
frwrd_band_centre_ solar_ elevation(i, k) < 0.

In this case use the d coefficient set.

sst = *d(zone*, *k*, 0) * 100. +
d(zone, *k*, 1) * I(ir11, n, i, j) +
d(zone, *k*, 2) * I(ir12, n, i, j) +
d(zone, *k*, 3) * I(ir37, n, i, j) +
d(zone, *k*, 4) * I(ir11, f, i, j) +
d(zone, *k*, 5) * I(ir12, f, i, j) +
d(zone, *k*, 6) * I(ir37, f, i, j)

```
combined_view_uses_ir37(i, j )  =  TRUE.
combined_image_valid(i, j )  =  NOT frwrd_cloud(i, j).
```

(Req 4.6-11)

If not using 3.7 micron channel the c coefficient set is required.

sst = *c(zone*, *k*, 0) * 100, +
c(zone, *k*, 1) * I(ir11, n, i, j) +
c(zone, *k*, 2) * I(ir12, n, i, j) +
c(zone, *k*, 3) * I(ir11, f, i, j) +
c(zone, *k*, 4) * I(ir12, f, i, j)

```
combined_view_uses_ir37(i, j )  =  FALSE.
combined_image_valid(i, j )  =  NOT frwrd_cloud(i, j).
```

(Req 4.6-12)

Step 4.6.6 Calculate the NDVI.

The NDVI is calculated for exended land pixels where the nadir view is not cloudy and both the .87 micron and .67 micron channels contain valid data.

extended_land(i, j) = TRUE AND
nadir_cloud(i, j) = FALSE AND
I(v870, n, i, j) > 0 AND
I(v670, n, i, j) > 0

Then

$$NDVI(i,j) = 10000\frac{I(v870,n;i,j) - I(v670,n;i,j)}{I(v870,n;i,j) + I(v670,n;i,j)}$$

and

```
combined_image_valid(i, j) = TRUE
```

(Req 4.6-13)

Otherwise set

$NDVI(i, j)$ = -19999

```
combined_image_valid(i, j) = FALSE
```

Step 4.6.7 Confidence flags

The following gsst confidence flags are set as appropriate:

```
land(i, j) = extended_land(i, j)
nadir_view_cloudy(i, j) =  nadir_cloud(i, j)
nadir_view_blanking(i, j) = nadir_blanking_pulse(i, j)
nadir_view_cosmetic(i, j) = nadir_cosmetic(i, j)

frwrd_view_cloudy(i, j) = frwrd_cloud(i, j)
frwrd_view_blanking(i, j) = frwrd_blanking_pulse(i, j)
frwrd_view_cosmetic(i, j) = frwrd_cosmetic(i, j)
                                                  (Req 4.6-14)
```

The following gsst confidence flags will have been set if the relevant calculations have been performed.

```
nadir_image_valid(i, j)
nadir_only_uses_ir37(i, j)
combined_image_valid(i, j)
combined_view_uses_ir37(i, j)
```

The following additional cloud flags are set on the basis of the input (GBTR) cloud flags:

If (NOT extended_land(i, j)) then

```
gsst_v16_cloud_test(i, j) = {nadir_v16_histogram_test(i, j) or
        nadir_v16_spatial_coherence_test(i, j) or
        frwrd_v16_histogram_test(i, j) or
        frwrd_v16_spatial_coherence_test(i, j)}
gsst_nadir_frwrd_cloud_test(i, j) = nadir_ir11_ir12_view_diff_test(i, j)
gsst_ir11_histogtram_test(i, j) = {nadir_ir11_ir12_histogram_test(i, j) or
        frwrd_ir11_ir12_histogram_test(i, j)}
                                                  (Req 4.6-15)
```

If the extended land flag is set, these flags will have been set in Step 4.6.9.

Step 4.6.8 Smoothing

Smoothing is applied to both the nadir and combined view SST images. The quantity that is smoothed is the atmospheric correction, defined as the difference between the derived sea-surface temperature and the 11 micron brightness temperature. The smoothing method is a Smooth_fac by Smooth_fac boxcar with checks for valid data and array bounds.

The 11 micron brightness temperature is then added to obtain the final smoothed SST.

Step 4.6.8.1 Smooth Nadir Image

For each pixel *i, j* in the nadir image the smoothed value is calculated as follows.

First initialize the output array element to the data invalid value.

```
smoothed_gsst_image(i, j )  =  -1.                (Req 4.6-16)
```

If the pixel contains a valid SST value ie :

nadir_image_valid(i, j) = TRUE   AND
nadir_view_cloudy(i, j) = FALSE   AND
extended_land(i, j) = FALSE

then calculate the average atmospheric correction using all the valid SST values in a smooth_fac by Smooth_fac pixel box centred on this pixel. ib = i - 1 to i + 1, and jb = j - 1 to j + 1.

The pixel (ib, jb) in the box has a valid SST value if

jb ≥ 0 AND jb < 512 AND
ib ≥ 0 AND ib < nadir_image_size AND
nadir_image_valid(ib, jb) = TRUE AND
nadir_view_cloudy(ib, jb) = FALSE AND
extended_land(ib, jb) = FALSE

If there is at least one valid pixel in the box use these valid pixels to calculate the average value of

nadir_image_field(ib, jb) - I(ir11, n; ib, jb)

then add I(ir11, n; i, j) to the average and store the result in

`smoothed_gsst_image(i, j).` (Req 4.6-17)

When all the averages have been calculated the valid values in the smoothed image can be copied to the image field array.

IF smoothed_gsst_image(i, j) > 0 THEN
nadir_image_field(i, j) = smoothed_gsst_image(i, j)

(Req 4.6-18)

Step 4.6.8.2 Smooth Combined Image

The process is repeated for the combined image field. For each pixel i, j in the combined image the smoothed value is calculated as follows. First initialize the output array element to the data invalid value.

`smoothed_gsst_image(i, j ) = -1.` (Req 4.6-19)

If the pixel contains a valid SST value ie :

combined_image_valid(i, j) = TRUE AND
nadir_view_cloudy(i, j) = FALSE AND
extended_land(i, j) = FALSE

then calculate the average atmospheric correction using all the valid SST values in a smooth_fac by Smooth_fac pixel box centred on this pixel. ib = i - 1 to i + 1, and jb = j - 1 to j + 1.

The pixel (ib, jb) in the box has a valid SST value if

jb ≥ 0 AND jb < 512 AND
ib ≥ 0 AND ib < nadir_image_size AND
combined_image_valid(ib, jb) = TRUE AND
nadir_view_cloudy(ib, jb) = FALSE AND
extended_land(ib, jb) = FALSE

If there is at least one valid pixel in the box use these valid pixels to calculate the average value of

nadir_image_field(ib, jb) - I(ir11, n; ib, jb)

then add I(ir11, n; i, j) to the average and store the result in

`smoothed_gsst_image(i, j).` (Req 4.6-20)

When all the averages have been calculated the valid values in the smoothed image can be copied to the image field array.

IF smoothed_gsst_image(i, j) > 0 THEN
combined_image_field(i, j) = smoothed_gsst_image(i, j)

(Req 4.6-21)

Step 4.6.9 Derive Land Surface Temperature Image.

LST retrievals use the nadir view 11 and 12 micron channels in conjunction with retrieval coefficients derived from the tables.

If either the 11 or 12 micron brightness temperature in the nadir view is invalid, the calculation for that pixel is abandoned and the nadir_image_valid flag remains *FALSE*. In this case set

```
nadir_image_field(i, j) = I(ir11,n; i, j).
```

(Req 4.6-22)

Otherwise the calculation proceeds as follows.

Step 4.6.9.1 Determine latitude and longitude indices

lat_index = integer part of [(*image_latitude*(*i*, *j*) + 90.0) × 2.0]

lon_index = integer part of [(*image_longitude*(*i*, *j*) + 180.0) × 2.0]

(Req 4.6-23)

disp_lat_index = integer part of [360 + (*image_latitude*(*i*, *j*) + 90.0) × 2.0 - 0.5] (modulo 360)

disp_lon_index = integer part of [720 + (*image_longitude*(*i*, *j*) + 180.0) × 2.0 - 0.5] (modulo 720)

(Req 4.6-24)

Step 4.6.9.2 Determine solar elevation and day/night flag

sun_elev = *nadir_band_centre_solar_elevation*(*i*, *band*(*j*))

If *sun_elev* > 0.0 then

night = 0 otherwise *night* = 1

(Req 4.6-25)

Step 4.6.9.3 Determine satellite elevation and non-linear exponent

A linear interpolation may be used to determine the satellite elevation.

w = *float*(*j* – 6)/50.0 - *band*(*j*)

sat_elev = (1.0 – *w*) × *nadir_band_edge_satellite_elevation*(*i*, *band*(*j*)) +

w × *nadir_band_edge_satellite_elevation*(*i*, *band*(*j*) + 1)

(Req 4.6-26)

if I(ir11, n; i, j) > I(ir12, n; i, j) then ***n*** = 1.0 / cos(π × (90 - *sat_elev*) / (*m* ×180.0))

else ***n*** = 1.0

(Req 4.6-27)

Note that *m* is [L2-AUX10-2] and ***n*** is [L2-INT-480].

Step 4.6.9.4 Determine coefficients

f = 0.001 × *vegetation_fraction*(*lat_index*, *lon_index*)

(Req 4.6-28)

Interpolation of precipitable water:

pw00 = *precipitable_water*(*disp_lat_index*, *disp_lon_index*)

pw01 = *precipitable_water*(*disp_lat_index+1*, *disp_lon_index*)

pw10 = *precipitable_water*(*disp_lat_index*, [*disp_lon_index+1*](modulo 720))

pw11 = *precipitable_water*(*disp_lat_index+1*, [*disp_lon_index+1*](modulo 720))

(Req 4.6-29)

q = fractional part of [(*image_latitude*(i, j) + 90.0) × 2.0 + 0.5]

p = fractional part of [(*image_longitude*(i, j) + 180.0) × 2.0 + 0.5]

$pw = 0.001 \times ((1 - p)(1 - q)pw00 + (1 - p)q \times pw01 + p(1 - q)pw10 + pq \times pw11)$

(Req 4.6-30)

class= *vegetation_class*(*lat_index*, *lon_index*) - 1

(Req 4.6-31)

If *class* < 0 or *class* > *NCLASS* **-** 1 then the index is out of range; the calculation for this pixel is abandoned and the nadir_image_valid flag remains false. In this case set

```
nadir_image_field(i, j) = I(ir11,n; i, j).
```

(Req 4.6-32)

Otherwise:

for k = 0, 2

$a(k) = f \times coeff(class, 0, night, k) + (1.0 - f) \times coeff(class, 1, night, k)$

(Req 4.6-33)

If (*class* + 1) = 14 this is an inland lake pixel. The exponent ***n*** and the preipitable water correction are not used. In this case set

n = 1.0.

(Req 4.6-34)

Otherwise, if (*class* + 1) ≠ 14 correct $a(0)$ as follows:

$a(0) = a(0) + d \times (cosec(\pi \times sat\_elev / 180.0) - 1.0) \times pw$

(Req 4.6-35)

Note that d is [L2-AUX10-1].

Step 4.6.9.5 Calculate the land surface temperature.

lst = 100. × (a(0) +
a(1) × (0.01 × (I(ir11, n; i, j) - I(ir12, n; i, j)))****n**) +
(a(1) + a(2)) × (I(ir12, n; i, j) -27315) + 27315

(Req 4.6-36)

topographic_variance(i, j) = *topographic_flag*(*lat_index*, *lon_index*)

(Note that this is a two-bit flag.)

(Req 4.6-37)

Trap for lst out of range:

If *lst* ≥ 32767.5 then

```
nadir_image_field(i, j) = 32767
nadir_image_valid(i, j) = FALSE.                        (Req 4.6-38)
```

else

```
nadir_image_field(i, j) = integer part of (lst + 0.5)
nadir_image_valid(i, j) = TRUE.                         (Req 4.6-39)
```

Step 4.6.9.6 Update 'marginal cloud' flags

For land pixels, the 'cloud test' flags have a different meaning. Note that for true land (excluding lake) pixels, these flags should be clear because the relevant tests are not applied to land pixels at Level 1B.

```
gsst_v16_cloud_test(i, j) = nadir_cloud(i, j)
gsst_nadir_frwrd_cloud_test(i, j) = NOT land(i, j)
gsst_ir11_histogtram_test(i, j) = FALSE
                                                        (Req 4.6-40)
```

The first of these is the actual 'marginal cloud' flag; it is set if the LST has been computed even though the nadir view pixel was flagged as cloudy. The second acts in practice as a lake flag, while the third is cleared (it will only have been set over an inland lake) for the avoidance of ambiguity.

Implementation Note: Land Surface temperature

Layout of auxiliary files

In this document the latitude and longitude of a pixel [*i*, *j*] are represented by φ(i, j), λ(i, j) respectively, with the conventions that

$$-90.0 \le \varphi \le +90.0 \text{ and}$$

$$-180.0 \le \lambda < 180.0$$

It will be convenient in the following to redefine the origin of latitude and longitude so that both are positive. We thus define the shifted co-ordinates

$$\hat{\varphi} = \varphi + 90.0$$
$$\hat{\lambda} = \lambda + 180.0$$

The auxiliary files define the surface class and vegetation fraction with a resolution of 0.5 degrees. Thus for each cell of 0.5 degrees in latitude by 0.5 degrees in longitude a surface class and vegetation fraction are defined, that are taken to apply to the whole cell.

Suppose that each cell is identified by the co-ordinates of its origin, defined to be its lower left-hand (i.e. south-west) corner. The cells form a two dimensional array indexed by latitude and longitude indices lat_index and lon_index, such that the origin of the cell indexed by lat_index and lon_index is

$$\hat{\varphi}_0 = lat\_index \times \Delta\varphi$$
$$\hat{\lambda}_0 = lon\_index \times \Delta\lambda$$

where $\Delta\varphi$, $\Delta\lambda$ are the cell dimensions in latitude and longitude respectively. It follows that a pixel (i, j) at latitude $\varphi(i, j)$, longitude $\lambda(i, j)$ falls within the cell identified by

$$lat\_index = \mathrm{int}\left[\frac{\hat{\varphi}}{\Delta\varphi}\right] = \mathrm{int}\left[\frac{\varphi(i,j)+90.0}{\Delta\varphi}\right],$$

$$lon\_index = \mathrm{int}\left[\frac{\hat{\lambda}}{\Delta\lambda}\right] = \mathrm{int}\left[\frac{\lambda(i,j)+180.0}{\Delta\lambda}\right].$$

where int[x] represents the integer part of x. In the present case

$\Delta\varphi = \Delta\lambda = 0.5$ degrees,

and so (compare Req 4.6-23)

$$lat\_index = \mathrm{int}[2\varphi(i,j)+180.0],$$
$$lon\_index = \mathrm{int}[2\lambda(i,j)+360.0].$$

For example, the cell at 58N, 7E extends over the latitude range 58.0 to 58.5 and the longitude range 7.0 to 7.5, and is indexed by lat_index = 296, lon_index = 374. The surface class for the cell will be found in the array element *vegetation_class*(*lat_index*, *lon_index*).

However, the precipitable water value is assumed to refer to the centre of the cell. Thus the precipitable water value associated with the cell [296][374] refers to the point at latitude 58.25 N, longitude 7.25 E. This shift must be taken into account in the interpolation of precipitable water. The centre points of the cells are the grid points for the bilinear interpolation of the precipitable water *pw*.

Interpolation of precipitable water

The water vapour sample corresponding to the cell whose origin is $\hat{\varphi}_0, \hat{\lambda}_0$ refers to the point whose shifted co-ordinates are $\hat{\varphi}_0 + \Delta\varphi/2, \hat{\lambda}_0 + \Delta\lambda/2$, and so the water vapour samples form a grid whose origin is at the point $\Delta\varphi/2$, $\Delta\lambda/2$. The precipitable water is interpolated to the position of the pixel using a bilinear interpolation between the four points of this grid that surround the pixel. These are the corner points of a quadrilateral enclosing the pixel. The origin of this quadrilateral is

$$\hat{\varphi}_0 = \mathrm{int}\left[\frac{\hat{\varphi}-\Delta\varphi/2}{\Delta\varphi}\right]\Delta\varphi + \Delta\varphi/2$$

$$\hat{\lambda}_0 = \mathrm{int}\left[\frac{\hat{\lambda}-\Delta\lambda/2}{\Delta\lambda}\right]\Delta\lambda + \Delta\lambda/2$$

and it clearly falls within the cell whose indices are

$$disp\_lat\_index = \mathrm{int}\left[\frac{\hat{\varphi}-\Delta\varphi/2}{\Delta\varphi}\right]$$

$$disp\_lon\_index = \mathrm{int}\left[\frac{\hat{\lambda}-\Delta\lambda/2}{\Delta\lambda}\right]$$

This pair of equations with $\Delta\varphi = \Delta\lambda = 0.5$ is equivalent to the pair of equations Req 4.6-24 except that we have modified the latter so that they always give a positive result. The indices of the four sample points that enter into the interpolation are (ignoring wrap-around)

disp_lat_index, *disp_lon_index*

disp_lat_index, *disp_lon_index* + 1

disp_lat_index + 1, *disp_lon_index*

disp_lat_index + 1, *disp_lon_index* + 1

The relationship of *disp_lat_index*, *disp_lon_index* to *lat_index*, *lon_index* depends in which quadrant of the cell the pixel lies. In the case that $\Delta\varphi = \Delta\lambda = 0.5$, the fractional co-ordinates of the pixel relative to the origin of the cell in which it falls are *cell_lat_coord*, *cell_long_coord* given by

$$cell\_lat\_coord = (2\varphi(i,j) + 180.0) - lat\_index$$
$$cell\_long\_coord = (2\lambda(i,j) + 360.0) - lon\_index$$

Both *cell_lat_coord*, *cell_long_coord* lie in the range [0, 1]. Then it follows from the equations for *disp_lat_index*, *disp_lon_index* that the indices (in the precipitable water array) of the origin of the interpolation quadrilateral are as follows:

If *cell_lat_coord* < 0.5 then *disp_lat_index* = *lat_index* - 1 else *disp_lat_index* = *lat_index*

If *cell_long_coord* < 0.5 then *disp_lon_index* = *lon_index* - 1 else *disp_lon_index* = *lon_index*.

The approach to the precipitable water interpolation actually adopted in the Reference Processor is as follows. The same set of *pw* sample values can be used for a succession of pixels, and an auxiliary 3 by 3 array is defined, to hold the precipitable water values of the current cell (*lat_index*, *lon_index*) and its eight neighbours. If this array is *pw_coeff*(*iy*, *jx*) then the array element *pw_coeff*(1, 1) contains the current cell value, and generally for *iy = 0, 1, 2* and *jx = 0, 1, 2*:

pw_coeff(*iy*, *jx*) = *precipitable_water*(*lat_index* +*iy* - 1, [*lon_index* + *jx* - 1](modulo 720))

The fractional co-ordinates of the pixel within the cell are *cell_lat_coord*, *cell_long_coord* as above, both in the range [0, 1]. Then the indices in this array of origin of the interpolation quadrilateral are as follows:

If *cell_lat_coord* < 0.5 then *iy* = 0 else *iy* = 1

If *cell_long_coord* < 0.5 then *jx* = 0 else *jx* = 1

It is easy to verify that this is equivalent to the formulation in terms of disp_lat_index, disp_lon_index given above. We then have

pw00 = *pw_coeff*(*iy*, *jx*)

pw01 = *pw_coeff*(*iy + 1*, *jx*)

pw10 = *pw_coeff*(*iy*, *jx+1*)

pw11 = pw_coeff(iy + 1, jx + 1)

Thus the procedure is as follows. For each pixel, its latitude and longitude are computed, and the indices *lat_index*, *lon_index* of the cell in which it falls are calculated. If this is the first pixel, or if the computed cell indices differ from those of the last pixel (so it falls in a new cell), the precipitable water values of the centre cell and of its eight neighbours are copied into the 3 by 3 array. The interpolation then proceeds as above.

4.7 Module Definition: Output GSST Records

4.7.1 Functional Description

The GSST product is written to the output medium. First the SPH and ADS records are output, then for each image line, an MDS record is assembled and written.

4.7.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
GBTR-ADS1-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS1-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS1-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS1-4		image scan y coordinate	sl	m	4	1
GBTR-ADS1-5		instrument scan number, nadir view	us	none	2	512
GBTR-ADS1-6		pixel number, nadir view	us	none	2	512
GBTR-ADS2-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS2-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS2-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS2-4		image scan y coordinate	sl	m	4	1
GBTR-ADS2-5		instrument scan number, forward view	us	none	2	512
GBTR-ADS2-6		pixel number, forward view	us	none	2	512
GBTR-ADS3-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS3-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS3-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS3-4		image scan y coordinate	sl	m	4	1
GBTR-ADS3-5		tie point latitudes	sl	mdeg	4	23
GBTR-ADS3-6		tie point longitudes	sl	mdeg	4	23
GBTR-ADS3-7		latitude corrections	ss	mdeg	2	23
GBTR-ADS3-8		longitude corrections	ss	mdeg	2	23
GBTR-ADS3-9		Topographic Altitude	ss	metres	2	23
GBTR-ADS4-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS4-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS4-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS4-4		instrument scan number	us	none	2	1
GBTR-ADS4-5		tie pixel x coordinate	sl	m	4	94
GBTR-ADS4-6		tie pixel y coordinate	sl	m	4	94
GBTR-ADS5-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS5-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GBTR-ADS5-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS5-4		image scan y coordinate	sl	m	4	1
GBTR-ADS5-5		tie point solar elevation, nadir view	sl	mdeg	4	11
GBTR-ADS5-6		tie point satellite elevation, nadir view	sl	mdeg	4	11
GBTR-ADS5-7		tie point solar azimuth, nadir view	sl	mdeg	4	11
GBTR-ADS5-8		tie point satellite azimuth, nadir view	sl	mdeg	4	11
GBTR-ADS6-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GBTR-ADS6-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1

GBTR-ADS6-3		Spare (null characters)	3*uc	n/a	3	1
GBTR-ADS6-4		image scan y coordinate	sl	m	4	1
GBTR-ADS6-5		tie point solar elevation, forward view	sl	mdeg	4	11
GBTR-ADS6-6		tie point satellite elevation, forward view	sl	mdeg	4	11
GBTR-ADS6-7		tie point solar azimuth, forward view	sl	mdeg	4	11
GBTR-ADS6-8		tie point satellite azimuth, forward view	sl	mdeg	4	11

Table 4.7.1: Input Data Table - Output GSST Records

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-INT-270		nadir_image_field(i, j)	ss	0.01 K	2	j = 0, 511
L2-INT-271		combined_image_field(i, j)	ss	mixed	2	j = 0, 511
L2-INT-272		gsst_confidence_word(i, j)	ss	flags	2	j = 0, 511
		gsst confidence flags				
L2-INT-280		nadir_only_sst_valid(i, j)	flag	n/a	2	j = 0, 511
L2-INT-281		nadir_only_sst_uses_ir37(i, j)	flag	n/a	2	j = 0, 511
L2-INT-282		dual_view_sst_valid(i, j)	flag	n/a	2	j = 0, 511
L2-INT-283		dual_view_sst_uses_ir37(i, j)	flag	n/a	2	j = 0, 511
L2-INT-284		land(i, j)	flag	n/a	2	j = 0, 511
L2-INT-285		nadir_view_cloudy(i, j)	flag	n/a	2	j = 0, 511
L2-INT-286		nadir_view_blanking_pulse(i, j)	flag	n/a	2	j = 0, 511
L2-INT-287		nadir_view_cosmetic(i, j)	flag	n/a	2	j = 0, 511
L2-INT-288		frwrd_view_cloudy(i, j)	flag	n/a	2	j = 0, 511
L2-INT-289		frwrd_view_blanking_pulse(i, j)	flag	n/a	2	j = 0, 511
L2-INT-290		frwrd_view_cosmetic(i, j)	flag	n/a	2	j = 0, 511
L2-INT-291		gsst_v16_cloud_test(i, j)	flag	n/a	2	j = 0, 511
L2-INT-292		gsst_nadir_frwrd_cloud_test(i, j)	flag	n/a	2	j = 0, 511
L2-INT-293		gsst_ir11_histogtram_test(i, j)	flag	n/a	2	j = 0, 511

Table 4.7.2: Internal Data Table - Output GSST Records

Output Data Tables

Parameter ID	Variable	Name	Type	Units	Field size	Fields
GSST-MDS1-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GSST-MDS1-2		Record Quality Indicator	sc	n/a	1	1
GSST-MDS1-3		Spare (null characters)	3*uc	n/a	3	1
GSST-MDS1-4		image scan y coordinate	sl	m	4	1
GSST-MDS1-5		confidence words	us	flags	2	512
GSST-MDS1-6		nadir field	ss	K/100	2	512
GSST-MDS1-7		combined field (Note 1)	ss	K/100	2	512
		ADS:				
GSST-ADS1-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GSST-ADS1-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GSST-ADS1-3		Spare (null characters)	3*uc	n/a	3	1
GSST-ADS1-4		image scan y coordinate	sl	m	4	1
GSST-ADS1-5		instrument scan number, nadir view	us	none	2	512
GSST-ADS1-6		pixel number, nadir view	us	none	2	512
GSST-ADS2-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GSST-ADS2-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GSST-ADS2-3		Spare (null characters)	3*uc	n/a	3	1
GSST-ADS2-4		image scan y coordinate	sl	m	4	1
GSST-ADS2-5		instrument scan number, forward view	us	none	2	512
GSST-ADS2-6		pixel number, forward view	us	none	2	512
GSST-ADS3-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GSST-ADS3-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1

GSST-ADS3-3		Spare (null characters)	3*uc	n/a	3	1
GSST-ADS3-4		image scan y coordinate	sl	m	4	1
GSST-ADS3-5		tie point latitudes	sl	mdeg	4	23
GSST-ADS3-6		tie point longitudes	sl	mdeg	4	23
GSST-ADS3-7		latitude corrections	ss	mdeg	2	23
GSST-ADS3-8		longitude corrections	ss	mdeg	2	23
GSST-ADS3-9		Topographic Altitude	ss	metres	2	23
GSST-ADS4-1		Scan UTC time in MJD format	sl, 2*ul	MJD	12	1
GSST-ADS4-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GSST-ADS4-3		Spare (null characters)	3*uc	n/a	3	1
GSST-ADS4-4		instrument scan number	us	none	2	1
GSST-ADS4-5		tie pixel x coordinate	sl	m	4	94
GSST-ADS4-6		tie pixel y coordinate	sl	m	4	94
GSST-ADS5-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GSST-ADS5-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GSST-ADS5-3		Spare (null characters)	3*uc	n/a	3	1
GSST-ADS5-4		image scan y coordinate	sl	m	4	1
GSST-ADS5-5		tie point solar elevation, nadir view	sl	mdeg	4	11
GSST-ADS5-6		tie point satellite elevation, nadir view	sl	mdeg	4	11
GSST-ADS5-7		tie point solar azimuth, nadir view	sl	mdeg	4	11
GSST-ADS5-8		tie point satellite azimuth, nadir view	sl	mdeg	4	11
GSST-ADS6-1		Nadir UTC time in MJD format	sl, 2*ul	MJD	12	1
GSST-ADS6-2		Attachment flag (always zero for this ADS)	sc	n/a	1	1
GSST-ADS6-3		Spare (null characters)	3*uc	n/a	3	1
GSST-ADS6-4		image scan y coordinate	sl	m	4	1
GSST-ADS6-5		tie point solar elevation, forward view	sl	mdeg	4	11
GSST-ADS6-6		tie point satellite elevation, forward view	sl	mdeg	4	11
GSST-ADS6-7		tie point solar azimuth, forward view	sl	mdeg	4	11
GSST-ADS6-8		tie point satellite azimuth, forward view	sl	mdeg	4	11

Table 4.7-3: Output Data Table - Output GSST Records

Note 1. The GSST product is switchable, so that the contents of these MDS fields depend on the setting of the forward and nadir cloud flags and the land flag. The units and range of these quantities consequently depend on the flag settings.

4.7.3 Detailed Structure

Step 4.7.1 SPH record.

The SPH record is identical to that of the input (Level 1b) product. The SPH (excluding the Data Set Descriptors) should be read and copied unchanged to the output header. Suitable DSD records for the ADS and MDS defined below should be prepared and appended to the SPH of the level 2 product.

(Req 4.7-1)

Step 4.7.2 Ancillary Data sets.

The Ancillary Data sets ADS #0 to ADS #6 inclusive are identical to the corresponding Data Sets of the Level 1b product, with the exception of four fields of the SQ ADS (ADS #0); see Step 4.7.2.1 below. For each of GSST-ADS<n>, n= 1, 2, ...6, the corresponding records of GBTR-ADS<n> should be read and copied unchanged to the output data set GSST-ADS<n>.

(Req 4.7-2)

Step 4.7.2.1 Summary Quality ADS (ADS #0).

The Summary Quality ADS, ADS #0, shall be identical to the input Summary Quality ADS from the Level 1b (GBTR) Product, but with the addition of the new quantities [GSST-

ADS0-10], [GSST-ADS0-11], [GSST-ADS0-12] and [GSST-ADS0-13] (see [RD2] Table 6-4).

```
Initialise counters:
n=0
n_cloud = 0
n_land = 0
n_sea = 0
n_ndvi_inv = 0
n_nadir_inv = 0
n_dual_inv = 0
Let ig be the index corresponding to the current ADS record. The quantities
in the record indexed by ig relate to image rows indexed by
       i = 512 * ig + k, k=0, 511. Then for each output record ig:
for k = 0, 511
       i = 511 * ig + k
for j = 0, 511
(Sums over 512 * 512 pixels:)
if (not unfilled) then
       n = n + 1
       n_cloud = ncloud + nadir_view_cloudy(i, j)
       if (land(i, j) AND !nadir_view_cloudy(i, j)) then
           n_land = n_land + 1
       end if
       if (land(i, j) AND !nadir_view_cloudy(i, j) AND
       !combined_image_valid(i, j)) then
           n_ndvi_inv = n_ndvi_inv + 1
       end if
       if (!land(i, j) AND !nadir_view_cloudy(i, j)) then
           n_sea = n_sea + 1
       end if
       if (!land(i, j) AND !nadir_view_cloudy(i, j) AND
       !nadir_image_valid(i, j) then
           n_nadir_inv = n_nadir_inv + 1
       end if
       if (!land(i, j) AND !nadir_view_cloudy(i, j) AND
       !combined_image_valid(i, j)) then
           n_dual_inv = n_dual_inv + 1
       end if
endif
end for (loop over j)
end for (loop over k)
```

Each sum above is the number of pixels in the image segment that have the specified property.

```
(percentage of cloudy pixels:)
[GSST-ADS0-10](ig) = (if n = 0 then 0
       else 100.*n_cloud/float(n))
(percentage of NDVI invalid:)
[GSST-ADS0-11](ig) = (if n_land = 0 then 0
       else 100.*n_ndvi_inv/float(n_land))
(percentage of SST (nadir view) invalid:)
[GSST-ADS0-12](ig) = (if n_sea = 0 then 0
       else 100.*n_nadir_inv/float(n_sea))
```

```
(percentage of SST (dual view) invalid:)
[GSST-ADS0-13](ig) = (if n_sea = 0 then 0
      else 100.*n_dual_inv/float(n_sea))
```

Step 4.7.3 Measurement Data Set.

The GSST product includes only one measurement data set. This is to be assembled as follows. For each image scan i:

[GSST-MDS1-1](i) = [GBTR-MDS1-1](i)

[GSST-MDS1-2](i) = [GBTR-MDS1-2](i)

[GSST-MDS1-3](i) = [GBTR-MDS1-3](i)

[GSST-MDS1-4](i) = [GBTR-MDS1-4](i)

Assemble confidence words for each pixel j = 0, 511:

For each GSST confidence flag, the corresponding bit of the gsst confidence word is to be set according to the truth value (1 = TRUE; 0 = FALSE) of the flag as follows:

[gsst_confidence(i, j)](bit 0) = nadir_only_sst_valid(i, j)

[gsst_confidence(i, j)](bit 1) = nadir_only_sst_uses_ir37(i, j)

[gsst_confidence(i, j)](bit 2) = dual_view_sst_valid(i, j)

[gsst_confidence(i, j)](bit 3) = dual_view_sst_uses_ir37(i, j)

[gsst_confidence(i, j)](bit 4) = land(i, j)

[gsst_confidence(i, j)](bit 5) = nadir_view_cloudy(i, j)

[gsst_confidence(i, j)](bit 6) = nadir_view_blanking_pulse(i, j)

[gsst_confidence(i, j)](bit 7) = nadir_view_cosmetic(i, j)

[gsst_confidence(i, j)](bit 8) = frwrd_view_cloudy(i, j)

[gsst_confidence(i, j)](bit 9) = frwrd_view_blanking_pulse(i, j)

[gsst_confidence(i, j)](bit 10) = frwrd_view_cosmetic(i, j)

[gsst_confidence(i, j)](bit 11) = v16_cloud_test(i, j)

[gsst_confidence(i, j)](bit 12) = nadir_frwrd_cloud_test(i, j)

[gsst_confidence(i, j)](bit 13) = ir11_hist_cloud_test(i, j)

[gsst_confidence(i, j)](bit 14) = [topographic_variance(i, j)](bit 0)

[gsst_confidence(i, j)](bit 15) = [topographic_variance(i, j)](bit 1)

[GSST-MDS1-5](i, j) = gsst_confidence(i, j)

For each pixel j = 0, 511 [GSST-MDS1-6](i, j) = nadir_only_gsst_image(i, j)

For each pixel j = 0, 511 [GSST-MDS1-7](i, j) = dual_view_gsst_image(i, j)

(Req 4.7-3)

4.8 Module Definition: Spatial Averaging (Half Degree Cell)

4.8.1 Functional Description

For the averaged products in half-degree cells, the globe is divided into cells 0.5° in latitude by 0.5° in longitude, and these cells are further subdivided into 9 sub-cells extending 10 arcmin in latitude by 10 arcmin in longitude. For each channel, the average brightness temperature (for the infra-red channels) or reflectance (for the visible channels) is averaged over all pixels of each type that fall within each sub-cell, to give distributions of a brightness temperature and radiance at 10 arc minute resolution. Averages are performed for the forward and nadir views separately, and a separate average is performed for each surface type (land and sea) and cloud state (clear or cloudy). There are thus 4 averages per channel per view. The mean across-track band number in each cell is also derived, for use by the averaged SST algorithm.

4.8.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-AUX3-1		Threshold for ABT flag, 10 arcmin cell nadir view	sl	n/a	4	1
L2-AUX3-2		Threshold for ABT flag, 10 arcmin cell forward view	sl	n/a	4	1
L2-AUX3-3		Threshold for ABT flag, 30 arcmin cell nadir view	sl	n/a	4	1
L2-AUX3-4		Threshold for ABT flag, 30 arcmin cell forward view	sl	n/a	4	1
L2-AUX3-9	NGRANULE	Granule Size	sl	none	4	1

Table 4.8.1: Input Data Table - Spatial Averaging (Half Degree Cell)

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-INT-101	I(1, n; i, j)	nadir ir12 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-102	I(2, n; i, j)	nadir ir11 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-103	I(3, n; i, j)	nadir ir37 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-104	I(4, n; i, j)	nadir v16 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-105	I(5, n; i, j)	nadir v870 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-106	I(6, n; i, j)	nadir v670 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-107	I(7, n, i, j)	nadir v555 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-111	I(1, f; i, j)	forward ir12 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-112	I(2, f; i, j)	forward ir11 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-113	I(3, f; i, j)	forward ir37 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-114	I(4, f; i, j)	forward v16 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-115	I(5, f; i, j)	forward v870 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-116	I(6, f; i, j)	forward v670 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-117	I(7, f, i, j)	forward v555 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-100	nadir_fill_state(i, j)	nadir fill state indicator	byte	none	1	j = 0, 511
L2-INT-110	frwrd_fill_state(i, j)	forward fill state indicator	byte	none	1	j = 0, 511
L2-INT-232	nadir_land(i, j)	nadir view land flag	ss array	flag	2	j = 0, 511
L2-INT-233	nadir_cloud(i, j)	nadir view cloud flag	ss array	flag	2	j = 0, 511
L2-INT-248	frwrd_land(i, j)	forward view land flag	ss array	flag	2	j = 0, 511
L2-INT-249	frwrd_cloud(i, j)	forward view cloud flag	ss array	flag	2	j = 0, 511
L2-INT-160	image_latitude(i, j)	image pixel latitude	float	degrees	4	j = 0, 511
L2-INT-161	image_longitude(i, j)	image pixel longitude	float	degrees	4	j = 0, 511
L2-INT-124		nadir_band_centre_solar_elevation(i, k′)	float	degrees	4	k′= 0, 9

L2-INT-144		frwrd_band_centre_solar_elevation(i, k′)	float	degrees	4	k′= 0, 9
(local)	cell_latitude	pixel latitude transformed to cell units	float	cell	4	1
(local)	cell_longitude	pixel longitude transformed to cell units	float	cell	4	1
(local)	cell_latitude_index	cell latitude index	sl	cell	4	1
(local)	cell_longitude_index	cell longitude index	sl	cell	4	1
L2-INT-26	time(sg)	scan UTC	double	days	8	per sg
L2-INT-134	scn_nadir(ig, j)	nadir view instrument scan number	us	none	4	j = 0, 511
L2-INT-135	pxl_nadir(ig, j)	nadir view instrument pixel number	us	none	4	j = 0, 511
L2-INT-154	scn_frwrd(ig, j)	forward view instrument scan number	us	none	4	j = 0, 511
L2-INT-155	pxl_frwrd(ig, j)	forward view instrument pixel number	us	none	4	j = 0, 511
L2-INT-60	band(j)	across-track band number	sl	none	4	j = 0, 511
local	i	along-track scan index	sl	none	4	1
local	j	pixel index j = 0, ..., 511	sl	none	4	1
local	ch	channel identifier ch = 1, ..., 7	sl	none	4	1
	v	view identifier nadir / forward	sl	none	4	
	sf	surface type identifier (0 = sea, 1 = land)	sl	none	4	
	cl	cloud state identifier (0 = clear, 1 = cloud)	sl	none	4	
local	ig	along-track granule index	sl	none	4	1
local	s	index to instrument scans	sl	none	4	1
local	sg	index to ADS #4 records	sl	none	4	1
local	k	sub-cell number, k = 0, ..., 8	sl	none	4	
local	cell	cell identifier	sl	none	4	
L2-INT-30	utc(cell)	cell UTC	double	days	8	per cell
L2-INT-31	utc(k, cell)	sub-cell UTC	double	days	8	k = 0, 8
L2-INT-49	nadir_day(k, cell)	nadir view day/night flag	ss	flag	2	k = 0, 8
L2-INT-50	frwrd_day(k, cell)	forward view day/night flag	ss	flag	2	k = 0, 8
L2-INT-45	nadir_solar_el(k, cell)	nadir solar elevation for sub-cell	float	degrees	4	k = 0, 8
L2-INT-46	frwrd_solar_el(k, cell)	frwrd solar elevation for sub-cell	float	degrees	4	k = 0, 8
L2-INT-47	cell_lat(cell)	cell latitude	sl	μdeg	4	per cell
L2-INT-48	cell_long(cell)	cell longitude	sl	μdeg	4	per cell
L2-INT-32	sub_cell_lat(k, cell)	sub-cell latitude	sl	μdeg	4	k = 0, 8
L2-INT-33	sub_cell_long(k, cell)	sub-cell longitude	sl	μdeg	4	k = 0, 8
L2-INT-34	sub_cell_band(k, cell)	sub-cell across-track band	ss	none	2	
L2-INT-36	S(ch, v; sf, cl, k, cell)	sub-cell total, ch = 1, ..., 7	sl	n/a	4	
L2-INT-37	M(ch, v; sf, cl, k, cell)	sub-cell valid pixel count, ch = 1, ..., 7	ss	none	2	
L2-INT-38	A(ch, v; sf, cl, k, cell)	sub-cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	sl	0.001K	4	
L2-INT-39	A(ch, v; sf, cl, k, cell)	sub-cell reflectance average (for visible channels ch = 4, 5, 6, 7)	ss	0.01%	2	
L2-INT-40	$\tilde{M}$ (ch, v; sf, cl, cell)	cell pixel count, ch = 1, ..., 7	ss	none	2	
L2-INT-41	$\tilde{A}$ (ch, v; sf, cl, cell)	cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	sl	0.001K	4	
L2-INT-42	$\tilde{A}$ (ch, v; sf, cl, cell)	cell reflectance average (for visible channels ch = 4, 5, 6, 7)	ss	0.01%	2	
L2-INT-43	σ(ch, v; sf, cl, cell)	standard deviation of the cell average	float	0.001K or 0.01%	4	
L2-INT-355	N(v; sf, cl, k, cell)	sub-cell filled pixel count	ss	none	2	
L2-INT-356	band_sum(k, cell)	cumulative across-track band sum	sl	none	4	
L2-INT-357	mean_band(k, cell)	mean across-track band number	ss	none	2	
L2-INT-358	across_track_sum(sf; k, cell)	cumulative sum of across-track pixel index	sl	none	4	k = 0, 8
L2-INT-359	across_track_mean(sf; k, cell)	mean across-track pixel index, subcell k	ss	none	2	k = 0, 8
L2-INT-368	across_track_mean(sf; cell)	mean across-track pixel index, cell	ss	none	2	per cell
local	μ	number of sub-cells contributing to mean	sl	none	4	
L2-INT-51	PFF(v, sf; cell)	Pixel threshold failure flags word, 30 arc minute cell	ss	flags	2	
L2-INT-52	PFF(v, sf; k, cell)	Pixel threshold failure flags word, 10 arc minute sub-cell	ss	flags	2	

L2-INT-301	N_land(n; k, cell)	total filled pixels over land for subcell	ss	none	2	k = 0, 8
L2-INT-302	N_sea(n; k, cell)	total of filled pixels over sea for subcell	ss	none	2	k = 0, 8
L2-INT-303	N_total(n; k, cell)	total of filled pixels for subcell, nadir view	ss	none	2	k = 0, 8
L2-INT-304	pcs(n; k, cell)	percentage of cloudy pixels over sea	ss	0.01%	2	k = 0, 8
L2-INT-305	pcl(n; k, cell)	percentage of cloudy pixels over land	ss	0.01%	2	k = 0, 8
L2-INT-306	N_land(n; cell)	total filled pixels over land for cell	ss	none	2	
L2-INT-307	N_sea(n; cell)	total of filled pixels over sea for cell	ss	none	2	
L2-INT-308	N_total(n; cell)	total of filled pixels for cell, nadir view	ss	none	2	
L2-INT-309	pcs(n; cell)	percentage of cloudy pixels over sea	ss	0.01%	2	
L2-INT-310	pcl(n; cell)	percentage of cloudy pixels over land	ss	0.01%	2	
L2-INT-311	N_land(f; k, cell)	total filled pixels over land for sub-cell	ss	none	2	k = 0, 8
L2-INT-312	N_sea(f; k, cell)	total of filled pixels over sea for subcell	ss	none	2	k = 0, 8
L2-INT-313	N_total(f; k, cell)	total of filled pixels for subcell, frwd view	ss	none	2	k = 0, 8
L2-INT-314	pcs(f; k, cell)	percentage of cloudy pixels over sea	ss	0.01%	2	k = 0, 8
L2-INT-315	pcl(f; k, cell)	percentage of cloudy pixels over land	ss	0.01%	2	k = 0, 8
L2-INT-316	N_land(f; cell)	total filled pixels over land for cell	ss	none	2	
L2-INT-317	N_sea(f; cell)	total of filled pixels over sea for cell	ss	none	2	
L2-INT-318	N_total(f; cell)	total of filled pixels for cell, frwd view	ss	none	2	
L2-INT-319	pcs(f; cell)	percentage of cloudy pixels over sea	ss	0.01%	2	
L2-INT-320	pcl(f; cell)	percentage of cloudy pixels over land	ss	0.01%	2	
L2-INT-491	sub_cell_index(k, cell)	Along-track index representative of sub-cell	sl	none	4	k = 0, 8

Table: 4.8.2 Internal Data Table - Spatial Averaging (Half Degree Cell)

4.8.3 Detailed Structure

As previously noted, the globe is divided into cells 0.5° in latitude by 0.5° in longitude. Each cell can be identified by two numbers defined to be non-negative; the latitude index given by 2*(latitude + 90) and a longitude index 2*(longitude + 180). All internal variables associated with the averaged product algorithms are duplicated for each cell, and should be imagined to be virtually present in memory at all times for the purpose of algorithm definition. How this is to be implemented is not specified here.

Each cell is further subdivided into 9 sub-cells each extending 10 arc minutes in latitude by 10 arc minutes in longitude. The sub-cells within each cell are identified by an index in the range 0 to 8 as follows:

6	7	8
3	4	5
0	1	2

For the averaged channel values there are for each channel and for each view four cumulative sums, depending on surface type and cloud flag as follows:

sea, clear	sea, cloud
land, clear	land, cloud

To simplify the notation we define a surface type index *sf* = 0 (sea) or 1 (land) and a cloud state index *cl* = 0 (clear) or 1 (cloud).

For the purpose of indexing and identifying the AATSR channels, the following conventional numbering scheme will be adopted.

Channel	Symbol	Index (ch)
12 micron	ir12	1

11 micron	ir11	2
3.7 micron	ir37	3
1.6 micron	v16	4
0.870 micron	v870	5
0.670 micron	v670	6
0.55 micron	v555	7

Each average is defined by a sum of the form

$$A(ch,v;sf,cl,k,cell) = \left\{\sum I(ch,v;i,j)\right\} / M(ch,v;sf,cl,k,cell)$$

where *v* indicates the view, either *f* (forward) or *n* (nadir) and where the sum is over all valid pixels which fall within the cell, are filled (or not unfilled), and have the correct cloud/surface type flag. That is, the sum is over all values of *i* and *j* such that all four of the following conditions are satisfied;

the co-ordinates of the pixel indexed by *i* and *j* fall within the sub-cell;

<view>*_fill_state*(*i, j*) is NOT UNFILLED_PIXEL;

I(ch, v; i, j) is valid

surface type, cloud state, flags at *i* and *j* have the correct value.

A total of 56 sums and averages are calculated. A separate group of totals are calculated for each channel and view (nadir and forward), there being 14 channel/view combinations. For each combination of channel and view four totals are maintained, and 4 averages computed, corresponding to the four combinations of the cloud state and surface type flags.

In addition, the mean of the across-track band number is calculated for the clear sea pixels, for use in the Averaged SST determination, and a count of the numbers of pixels in each category is to be maintained.

Step 4.8.1 Derive channel totals for each cell

For each image scan *i*:

perform steps 4.8.1.1 to 4.8.1.7 for each pixel in the scan (pixel index *j* = 0, 511) unless the pixel is unfilled in both images.

Step 4.8.1.1 Identify cell number, subcell index and pixel state

cell_latitude = (*image_latitude(i, j)* +90)* 2

cell_longitude = (*image_longitude(i, j)* +180)* 2

cell_latitude_index = integer part of *cell_latitude*

cell_longitude_index = integer part of *cell_longitude*

Define the cell number *cell* as a function of *cell_latitude_index* and *cell_longitude_index*. The relationship between these quantities may be implementation-dependent; *cell* is required as an identifier.

The sub-cell index *k* is given by

k = 3 [3 (fractional part of *cell_latitude*)] + [3 (fractional part of *cell_longitude*)] where the inclusion of a quantity in square brackets implies that the integer part is to be taken.

If the pixel identified by (*i, j*) is the first pixel to fall within the cell *cell*, ensure that all counters and cumulative sums are initialised to zero as follows for each *ch* = 1, 7, *v* = *nadir*, *frwrd*, *sf* = 0, 1, *cl* = 0, 1:

S(ch, v; sf, cl, k, cell) = 0.0

M(ch, v; sf, cl, k, cell) = 0

N(v; sf, cl; k, cell) = 0

band_sum(k, cell) = 0, for each *k* = 0, 8

across-track_sum(sf; k, cell) = 0 for each *k* = 0, 8.

Also initialise the latitude and longitude of the sub-cells to exceptional values (in case the cell is intersected by the swath edge, and no pixels fall within a sub-cell):

sub_cell_lat(k, cell) = -399999999,
sub_cell_long(k, cell) = -399999999, for each *k* = 0, 8.

If the pixel identified by (*i, j*) is the first pixel to fall within this cell, assemble the cell geolocation and allied information as follows:

The time tag associated with the cell [L2-INT-30] is the instrument scan time of the nadir pixel which first falls within the cell. It is derived from the scan number associated with the pixel, and with the scan times from ADS #4.

First we identify the instrument scan number associated with the pixel (i.e. the number of the scan from which the pixel was regridded). This comes from ADS#1. Given i, j of the pixel compute

ig = integer part of (*i/NGRANULE*)

and extract

s = scn_nadir(ig, j) + (*i - NGRANULE*ig)*

= [L2-INT-134]*(ig, j)* + (*i - NGRANULE*ig)*

sg = integer part of [(*s - scan(0))/NGRANULE*]

[Note scan(0) = [L2-INT-27](0) = instrument scan number of the first record of ADS #4.)

If *sg* < 0, set *sg* = 0.

If *sg* ≥ *sg_max*, where *sg_max* is the number of the last record in ADS #4, so that *time*(*sg* + 1) and *scan*(*sg* + 1) do not exist, set

sg = *sg_max* - 1.

This is the index of the scan time from ADS#4 from which the time tag is taken. It has already been converted to processing format in Step 4.2.2. The converted time is

t0 = time(*sg*)

The time *t1* is derived similarly from the time tag of the subsequent record *sg* + 1.

t1 = time(*sg* + 1)

Linear interpolation is then used to derive the UTC in processing format.

utc(cell) = *t*0 + (*t*1 - *t*0)(*s* - *sg***NGRANULE*)/*(scan(sg* + 1*)- scan(sg))*

(Note that the denominator *(scan(sg* + 1*)- scan(sg))* is equal to *NGRANULE* unless *sg* = *sg_max* - 1.)

cell_lat(cell) = (*cell_latitude_index* - 180)*500000

cell_long(cell) = (*cell_longitude_index* - 360)*500000

Similarly if the pixel identified by (*i*, *j*) is the first pixel to fall within the sub-cell *k*, assemble the sub-cell time tag and geolocation information as follows. First derive the UTC in processing format, *utc*(*k*, *cell*), from *i* and *j* in exactly the same way as described above for the cell. Then

sub_cell_lat(k, cell) = integer part of (3**cell_latitude* -540)*500000/3

sub_cell_long(k, cell) = integer part of (3**cell_longitude* - 1080)*500000/3

sub_cell_band(k, cell) = *band(j)*

sub_cell_index(k, cell) = *i*

(This is used in the LST calculation, Section 4.10.)

nadir_solar_el(k, cell) = *nadir_band_centre_solar_elevation(i, band(j))*

frwrd_solar_el(k, cell) =*frwrd_band_centre_solar_elevation(i, band(j))*

If <view>*_band_centre_solar_elevation(i, band(j))* > 0.0 then

<view>*_day(k, cell)* = TRUE

otherwise

<view>*_day(k, cell)* = FALSE

where <view> = <*nadir* | *frwrd*>

(Req 4.8-1)

Step 4.8.1.2 Process nadir Pixels

Perform steps 4.8.1.3 and 4.8.1.4 to process the nadir pixels unless the nadir pixel is unfilled (i.e. unless *nadir_fill_state(i, j)* = UNFILLED_PIXEL).

Step 4.8.1.3 Identify the surface type and cloud state associated with the nadir pixel:

sf = 0 if nadir view land flag [L2-INT-232](*i*, *j*) = FALSE

sf = 1 if nadir view land flag [L2-INT-232](*i*, *j*) = TRUE

cl = 0 if nadir view cloud flag [L2-INT-233](*i*, *j*) = FALSE

cl = 1 if nadir view cloud flag [L2-INT-233](*i*, *j*) = TRUE

Increment the pixel counters associated with the cloud state and surface type just determined:

$$N(n; sf, cl, k, cell) \leftarrow N(n; sf, cl, k, cell) + 1$$

If *cl* = 0 then increment the cumulative across-track index as follows:

$$across\_track\_sum(sf;k,cell) \leftarrow across\_track\_sum(sf;k,cell) + j$$

(Req 4.8-2)

Step 4.8.1.4 Update nadir view channel totals

For each channel of the nadir view *ch* perform this step if the corresponding nadir pixel is valid (that is, if $I(ch, n; i, j) > 0$):

$$S(ch,n;sf,cl,k,cell) \leftarrow S(ch,n;sf,cl,k,cell) + I(ch,n;i,j)$$

$$M(ch,n;sf,cl,k,cell) \leftarrow M(ch,n;sf,cl,k,cell) + 1$$

Step 4.8.1.5 Process forward Pixels:

Perform steps 4.8.3.1.6 and 4.8.3.1.7 to process the forward pixels unless the forward pixel is unfilled (i.e. unless *frwrd_fill_state*(*i*, *j*) = UNFILLED_PIXEL).

Step 4.8.1.6 Identify the surface type and cloud state associated with the forward pixel:

sf = 0 IF forward view land flag [L2-INT-248](*i*, *j*) = FALSE

sf = 1 IF forward view land flag [L2-INT-248](*i*, *j*) = TRUE

cl = 0 IF forward view cloud flag [L2-INT-249](*i*, *j*) = FALSE

cl = 1 IF forward view cloud flag [L2-INT-249](*i*, *j*) = TRUE

Increment the pixel counters associated with the cloud state and surface type just determined:

$$N(f;sf,cl,k,cell) \leftarrow N(f;sf,cl,k,cell) + 1$$

(Req 4.8-3)

Step 4.8.1.7 Update forward view channel totals

For each channel of the forward view, perform the following steps if the corresponding forward pixel is valid (that is, if $I(ch, f; i, j) > 0$):

$$S(ch,f;sf,cl,k,cell) \leftarrow S(ch,f;sf,cl,k,cell) + I(ch,f;i,j)$$

$$M(ch,f;sf,cl,k,cell) \leftarrow M(ch,f;sf,cl,k,cell) + 1$$

Step 4.8.2 Derive average values

When no more pixels remain to be added to the cell, or at the end of the data set, compute the averages. The following equation is evaluated for each channel (*ch* = ir12, ir11, ir37, v16, v870, v670, v555), for each view *v* = *n*, *f*, for surface type *sf* = 0, 1 and for cloud state *cl* = 0, 1.

If $M(ch, v; sf, cl, k, cell) > 0$

$$A(ch,v;sf,cl,k,cell) = 10.0 \cdot S(ch,v;sf,cl,k,cell) / \mathrm{float}(M(ch,v;sf,cl,k,cell))$$

(note the conversion to units of 0.001 K) otherwise set

$$A(ch,v;sf,cl,k,cell) = -1.0$$

The mean in the larger (30 arc minute) cell is given by

$$\tilde{A}(ch,v;sf,cl,cell) = \frac{1}{\mu}\sum_k A(ch,v;sf,cl,k,cell) \text{ if } \mu > 0$$

$$\tilde{A}(ch,v;sf,cl,cell) = -1 \text{ if } \mu = 0$$

where the sum is over all $k \in \{0 \le k \le 8\}$ having a valid subcell mean *A* and μ is the number of such valid means. The number of pixels that contribute to the mean is similarly

$$\tilde{M}(ch,v;sf,cl,cell) = \sum_k M(ch,v;sf,cl,k,cell)$$

The standard deviation of the mean is

$$\sigma(ch,v;sf,cl,cell) = \left\{\frac{1}{\mu-1}\sum_k (A(ch,v;sf,cl,k,cell) - \tilde{A}(ch,v;sf,cl,cell))^2\right\}^{1/2}$$

provided μ > 1, otherwise set the standard deviation to –1. In all cases the sum is over sub-cells having valid means (i.e the number of contributing pixels *M* is positive).

(Req 4.8-4)

The mean across-track pixel number [L2-INT-359] is calculated for each sub-cell *k* = 0, 8 and for each surface type *sf* = 0, 1:

If *N(n; sf*, 0, *k, cell)* > 0 then set

across_track_mean(sf, k, cell) =

integer part of (*across_track_sum(sf; k, cell)* / *N(n; sf*, 0, *k, cell)*)

otherwise set

across_track_mean(sf, k, cell) = -1

The mean to be associated with the 30 arc minute cell is [L2-INT-368]

$$across\_track\_mean(sf;\, cell) = \frac{1}{\mu}\sum_k across\_track\_mean(sf;k,cell) \text{ if } \mu > 0$$

across_track_mean(sf; cell) = -1 if μ = 0

where the sum is over all $k \in \{0 \le k \le 8\}$ for which the number of contributing pixels

N(n; sf, 0, *k, cell)* > 0

and μ is the number of valid *k*.

(Req 4.8-5)

Step 4.8.3 Derive Pixel Threshold Failure Flags Words (10 arcminute cells)

For cell *cell* and for each sub-cell *k* = 0, 9:

For surface type *sf* = 0, 1 and for the nadir view v = n:

[*PFF(n, sf; k, cell)*](bit 0) = 1 if *M(ir12, n; sf, 0, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 1) = 1 if *M(ir11, n; sf, 0, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 2) = 1 if *M(ir37, n; sf, 0, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 3) = 1 if *M(v16, n; sf, 0, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 4) = 1 if *M(v870, n; sf, 0, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 5) = 1 if *M(v670, n; sf, 0, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 6) = 1 if *M(v555, n; sf, 0, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 7) = 1 if *M(ir12, n; sf, 1, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 8) = 1 if *M(ir11, n; sf, 1, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 9) = 1 if *M(ir37, n; sf, 1, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 10) = 1 if *M(v16, n; sf, 1, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 11) = 1 if *M(v870, n; sf, 1, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 12) = 1 if *M(v670, n; sf, 1, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 13) = 1 if *M(v555, n; sf, 1, k, cell)* < [L2-AUX3-1], otherwise 0

[*PFF(n, sf; k, cell)*](bit 14) = 1 if *nadir_day(k, cell)*= TRUE, otherwise 0

[*PFF(n, sf; k, cell)*](bit 15) = 0

(Req 4.8-6)

Similarly for surface type *sf* = 0, 1 and for the forward view *v* = *f*

Calculate the corresponding word *PFF(f, sf; k, cell)*:

Set bits 0 to 13 inclusive as above, substituting the view index *f* in place of *n*, and substituting the forward threshold value [L2-AUX3-2] in place of [L2-AUX3-1].

[*PFF(f, sf; k, cell)*](bit 14) = 1 if *frwrd_day(k, cell)*= TRUE, otherwise 0

[*PFF(f, sf; k, cell)*](bit 15) = 0

(Req 4.8-7)

Step 4.8.4 Derive Pixel Threshold Failure Flags Words (30 arcminute cells)

For cell *cell*:

For surface type *sf* = 0, 1 and for the nadir view v = n:

[*PFF(n, sf; cell)*](bit 0) = 1 if $\tilde{M}$ *(ir12, n; sf, 0, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 1) = 1 if $\tilde{M}$ *(ir11, n; sf, 0, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 2) = 1 if $\tilde{M}$ *(ir37, n; sf, 0, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 3) = 1 if $\tilde{M}$ *(v16, n; sf, 0, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 4) = 1 if $\tilde{M}$ *(v870, n; sf, 0, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 5) = 1 if $\tilde{M}$ *(v670, n; sf, 0, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 6) = 1 if $\tilde{M}$ *(v555, n; sf, 0, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 7) = 1 if $\tilde{M}$ *(ir12, n; sf, 1, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 8) = 1 if $\tilde{M}$ *(ir11, n; sf, 1, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 9) = 1 if $\tilde{M}$ *(ir37, n; sf, 1, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 10) = 1 if $\tilde{M}$ *(v16, n; sf, 1, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 11) = 1 if $\tilde{M}$ *(v870, n; sf, 1, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 12) = 1 if $\tilde{M}$ *(v670, n; sf, 1, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 13) = 1 if $\tilde{M}$ *(v555, n; sf, 1, cell)* < [L2-AUX3-3], otherwise 0

[*PFF(n, sf; cell)*](bit 14) = 1 if *nadir_day(k, cell)*= TRUE for some *k*, otherwise 0

[*PFF(n, sf; cell)*](bit 15) = 0

(Req 4.8-8)

Similarly for surface type *sf* = 0, 1 and for the forward view *v* = *f*

Calculate the corresponding word *PFF(f, sf; cell)*:

Set bits 0 to 13 inclusive as above, substituting the view index *f* in place of *n*, and substituting the forward threshold value [L2-AUX3-4] in place of [L2-AUX3-3].

[*PFF(f, sf; k, cell)*](bit 14) = 1 if *frwrd_day(k, cell)*= TRUE for some *k*, otherwise 0

[*PFF(f, sf; k, cell)*](bit 15) = 0

Step 4.8.5. Derive pixel counts for cell

For each cell *cell* and for each view *v* = *n, f*:

For each subcell *k* = 0, 8:

Total of filled pixels over land:

N_land(v; k, cell) = N(v; 1, 0, k, cell) + N(v; 1, 1, k, cell)

(Req 4.8-9)

Total of filled pixels over sea:

N_sea(v; k, cell) = N(v; 0, 0, k, cell) + N(v; 0, 1, k, cell)

(Req 4.8-10)

Total of filled pixels:

N_total(v; k, cell) = N_land(v; k, cell) + N_sea(v; k, cell)

(Req 4.8-11)

Derive cloudy pixel percentages for each subcell:

pcs(v; k, cell) = 10000 * *N(v; 0, 1, k, cell)/N_sea(v; k, cell)*

pcl(v; k, cell) = 10000 * *N(v; 1, 1, k, cell)/N_land(v; k, cell)*

(Req 4.8-12)

end for (*k*)

Derive aggregate counts:

Total of filled pixels over land:

$$N\_land(v; cell) = \sum_{k=0}^{8} N\_land(v; k, cell)$$

(Req 4.8-13)

Total of filled pixels over sea:

$$N\_sea(v; cell) = \sum_{k=0}^{8} N\_sea(v; k, cell)$$

(Req 4.8-14)

Total of filled pixels:

$$N\_total(v; cell) = N\_land(v; cell) + N\_sea(v; cell)$$

(Req 4.8-15)

Derive cloudy pixel percentages for the cell:

$$pcs(v; cell) = 10000 * (\sum_{k=0}^{8} N(v; 0, 1, k, cell))/N\_sea(v; cell)$$

$$pcl(v; cell) = 10000 * (\sum_{k=0}^{8} N(v; 1, 1, k, cell))/N\_land(v; cell)$$

(Req 4.8-16)

end for (*cell*, *v*)

4.9 Module Definition: Averaged SST Retrieval (Half Degree Cell)

4.9.1 Functional Description

This module derives the averaged SST from the averaged brightness temperatures determined using the module described in Section 4.8 above.

4.9.2 Interface Definition

Averaged SST Retrieval Coefficients

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-AUX2-1	a[i][j][0]	averaged sst retrieval a[3][38][0]	float	0.01K	4	114
L2-AUX2-2	a[i][j][1]	averaged sst retrieval a[3][38][1]	float	none	4	114
L2-AUX2-3	a[i][j][2]	averaged sst retrieval a[3][38][2]	float	none	4	114
L2-AUX2-4	b[i][j][0]	averaged sst retrieval b[3][38][0]	float	0.01K	4	114
L2-AUX2-5	b[i][j][1]	averaged sst retrieval b[3][38][1]	float	none	4	114
L2-AUX2-6	b[i][j][2]	averaged sst retrieval b[3][38][2]	float	none	4	114
L2-AUX2-7	b[i][j][3]	averaged sst retrieval b[3][38][3]	float	none	4	114
L2-AUX2-8	c[i][j][0]	averaged sst retrieval c[3][38][0]	float	0.01K	4	114
L2-AUX2-9	c[i][j][1]	averaged sst retrieval c[3][38][1]	float	none	4	114
L2-AUX2-10	c[i][j][2]	averaged sst retrieval c[3][38][2]	float	none	4	114

L2-AUX2-11	c[i][j][3]	averaged sst retrieval c[3][38][3]	float	none	4	114
L2-AUX2-12	c[i][j][4]	averaged sst retrieval c[3][38][4]	float	none	4	114
L2-AUX2-13	d[i][j][0]	averaged sst retrieval d[3][38][0]	float	0.01K	4	114
L2-AUX2-14	d[i][j][1]	averaged sst retrieval d[3][38][1]	float	none	4	114
L2-AUX2-15	d[i][j][2]	averaged sst retrieval d[3][38][2]	float	none	4	114
L2-AUX2-16	d[i][j][3]	averaged sst retrieval d[3][38][3]	float	none	4	114
L2-AUX2-17	d[i][j][4]	averaged sst retrieval d[3][38][4]	float	none	4	114
L2-AUX2-18	d[i][j][5]	averaged sst retrieval d[3][38][5]	float	none	4	114
L2-AUX2-19	d[i][j][6]	averaged sst retrieval d[3][38][6]	float	none	4	114
L2-AUX6-1	j	pixel index (j)	ss	none	2	512
L2-AUX6-2	map(j)	Across-track band index (map index)	ss	none	2	512
L2-AUX3-11		TROPICAL_INDEX (= 12.5)	float	degrees	4	1
L2-AUX3-12		TEMPERATE_INDEX (= 37)	float	degrees	4	1
L2-AUX3-13		POLAR_INDEX (= 70)	float	degrees	4	1
L2-AUX3-14		NADIR_PIXELS_THRESH	float	none	4	1
L2-AUX3-15		FRWRD_PIXELS_THRESH	float	none	4	1
L2-AUX3-16		IR37_THRESH	float	none	4	1

Table 4-9-1: Input Data Table - Averaged SST Retrieval Coefficients

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-INT-30	utc(cell)	cell UTC	double	days	8	per cell
L2-INT-31	utc(k, cell)	sub-cell UTC	double	days	8	k = 0, 8
L2-INT-32	sub_cell_lat(k, cell)	sub-cell latitude	sl	µdeg	4	k = 0, 8
L2-INT-33	sub_cell_long(k, cell)	sub-cell longitude	sl	µdeg	4	k = 0, 8
L2-INT-357	mean_band(k, cell)	mean across-track band number	ss	none	2	k = 0, 8
L2-INT-36	S(ch, v; sf, cl, k, cell)	sub-cell total, ch = 1, ..., 7	sl	n/a	4	
L2-INT-37	M(ch, v; sf, cl, k, cell)	sub-cell pixel count, ch = 1, ..., 7	ss	none	2	
L2-INT-38	A(ch, v; sf, cl, k, cell)	sub-cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	ss	0.01K	2	
L2-INT-40	$\tilde{M}$ (ch, v; sf, cl, cell)	cell pixel count, ch = 1, ..., 7	ss	none	2	
L2-INT-41	$\tilde{A}$ (ch, v; sf, cl, cell)	cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	ss	0.01K	2	
L2-INT-43	σ(ch, v; sf, cl, cell)	standard deviation of the cell average	float	0.01K or 0.01%	4	
L2-INT-49	nadir_day(k, cell)	nadir view sub-cell day/night flag	ss	flag	2	k = 0, 8
L2-INT-50	frwd_day(k, cell)	forward view sub-cell day/night flag	ss	flag	2	k = 0, 8
L2-INT-45	nadir_solar_el(k, cell)	nadir solar elevation for sub-cell	float	degrees	4	
L2-INT-46	frwd_solar_el(k_cell)	frwd solar elevation for sub-cell	float	degrees	4	
local	i	index to latitude zone; i = 0, 1, 2	sl	none	4	1
local	j	index to across-track bands j = 0, 9	sl	none	4	1
local	q	index to coefficient set	sl	none	4	1
L2-INT-359	across_track_mean(sf; k, cell)	mean across-track pixel index, subcell k	ss	none	2	k = 0, 8
L2-INT-360	a(i, j, q)	averaged sst retrieval a coefficients	float	mixed	4	342
L2-INT-361	b(i, j, q)	averaged sst retrieval b coefficients	float	mixed	4	456
L2-INT-362	c(i, j, q)	averaged sst retrieval c coefficients	float	mixed	4	570
L2-INT-363	d(i, j, q)	averaged sst retrieval d coefficients	float	mixed	4	798
L2-INT-366		nadir_asst_uses_ir37(k, cell)	ss	flag	2	k = 0, 8
L2-INT-367		dual_asst_uses_ir37(k, cell)	ss	flag	2	k = 0, 8
L2-INT-369	sst_mean_pixel(sf; cell)	mean across-track pixel index, cell	ss	none	2	per cell
	minpn	minimum nadir pixels	float	none	4	
	minpf	minimum forward pixels	float	none	4	
local	latitude	temporary latitude	float	degrees	4	
L2-INT-53	T_nadir(cell)	nadir view sst	ss	0.01K	2	
L2-INT-54	T_nadir(k, cell)	sub-cell nadir view sst	ss	0.01K	2	k = 0, 8
L2-INT-55	T_dual(cell)	dual view sst	ss	0.01K	2	

L2-INT-56	T_dual(k, cell)	sub-cell dual view sst	ss	0.01K	2	k = 0, 8
L2-INT-57	σ_nadir(ASST; cell)	standard deviation of nadir view ASST	ss	0.01K	2	
L2-INT-58	σ_dual(ASST; cell)	standard deviation of dual view ASST	ss	0.01K	2	
L2-INT-61	map(j)	across-track mapping	sl	none	4	j = 0, 511
L2-INT-364	ast_conf(0; k, cell)	AST confidence word for sea sub-cell	sl	flags	4	k = 0, 8
L2-INT-365	ast_conf(0; cell)	AST confidence word for sea cell	sl	flags	4	per cell
local	T0	tropical_sst	float	deg.	4	1
local	T1	temperate_sst	float	deg.	4	1
local	T2	polar_sst	float	deg.	4	1
local	w	interpolation weight	float	none	4	1
local	μ_1	number of sub-cells in cell average (nadir)	sl	none	4	
local	μ_2	number of sub-cells in cell average (dual)	sl	none	4	

Table 4-9-2: Internal Data Table - Averaged SST Retrieval (Half Degree Cell)

4.9.3 Detailed Structure

Both dual-view and nadir only sea surface temperatures are derived.

In the processing, each half-degree cell is represented by a structure containing, or is associated with, the necessary intermediate and output variables including the averaged brightness temperatures for each 10 arc-minute cell contained within the larger cell. All cells should be virtually present in memory, but how this is achieved is a matter for the implementer.

Note on notation: In the following we adopt the following abbreviated notation for the average brightness temperatures.

$$T_{ch}^{nadir} = \left[\frac{\text{float}(S(ch,n;0,0,k,cell))}{\text{float}(M(ch,n;0,0,k,cell))} \right]$$

$$T_{ch}^{frwrd} = \left[\frac{\text{float}(S(ch,f;0,0,k,cell))}{\text{float}(M(ch,f;0,0,k,cell))} \right]$$

where *ch* indicates one of the seven channels. This notation is adopted to reduce the proliferation of indices; note that where it is used, a dependence on *k* and *cell* is implied. Note also that the above quantities must be computed using a floating point computation, although *S* and *M* are of type integer, to ensure that sufficient precision is maintained. Substitution of the quantities *A*(*ch, f; 0, 0, k, cell*) would not ensure this.

This processing is applied to cells when the processing of Step 4.8.1 is complete; i.e. no more pixels remain to be added to the cell.

The processing to derive averaged SST is done as follows:

Step 4.9.1 Read look-up tables.

On first entry, input the look-up tables of averaged SST retrieval coefficients.

This is done once at initialisation. Retrieval coefficients are specified for three latitude zones (tropical, temperate and polar) and for 38 bands or strips running parallel to the ground track, and corresponding to different viewing angles. Distinct sets of coefficients are supplied for day/night and for nadir only/dual view retrievals, as follows.

Index	Zone	Set	Application

0	tropical	a	nadir only, day
1	temperate	b	nadir only, night
2	polar	c	dual view, day
		d	dual view, night

Before reading the retrieval coefficients ensure that the mapping array *map*(*j*) [L2-INT-61] is available. This has been read in during Step 4.6.1. If it is desired to re-input it independently in this module then proceed as before. Open the data set L2-AUX6 and set

map(j) = [L2-AUX6-2](j), j = 0, 511

(Note that [L2-AUX6-1](j) = j, j = 0, 511.)

Open the file of retrieval coefficients to access data set L2-AUX2.

For each latitude zone *i* = 0, 1, 2 (outer loop) and for each across-track band *j* = 0 to 37 (inner loop);

Read in next record of file.
Extract the *a* coefficients [L2-INT-360] as follows
a(i, j, 0) = [L2-AUX2-1]
a(i, j, 1) = [L2-AUX2-2]
a(i, j, 2) = [L2-AUX2-3]

Similarly extract in the *b*, *c* and *d* sets of coefficients:
b(i, j, q) = [L2-AUX2-<4 + q>], *q* = 0, 1, 2, 3;
c(i, j, q) = [L2-AUX2-<8 + q>], *q* = 0, 1, 2, 3 4;
d(i, j, q) = [L2-AUX2-<13 + q>], *q* = 0, 1, 2, 3, 4, 5, 6.

(Req 4.9-1)

Step 4.9.2 Nadir view average.

Calculate the nadir view averaged SST value for each of the 10-arcmin cells. Note that in the following, if the flags *nadir_asst_uses_ir37*, *dual_asst_uses_ir37 are initialised to the value FALSE, then Reqs. 4-9-2a, 4-9-9a are logivally redundant.*

Step 4.9.2.1

Determine the minimum number of pixels required for the cell, for the nadir view, using the latitude value representative of the cell. This is (in degrees)

$$latitude = sub\_cell\_lat(k, cell) * 10^{-6}$$

The latitude dependent threshold is

$$minpn = 340 * NADIR\_PIXELS\_THRESH * \cos((\pi/180.)*latitude) + 1.$$

If $M(ir12, n; 0, 0, k, cell) \geq minpn$ and $M(ir11, n; 0, 0, k, cell) \geq minpn$ proceed to calculate the retrieved sst as below, otherwise set

$$T\_nadir(k, cell) = -1.0$$

(Req 4.9-2)

$$nadir\_asst\_uses\_ir37(k, cell) = FALSE$$

(Req 4-9-2a)

Step 4.9.2.2

For night-time data, if *nadir_day(k, cell)* = FALSE, test whether the ratio of pixels with valid 3.7 µm data is greater or less than the threshold value and use the appropriate (three or two channel) SST algorithm. The 3.7 micron channel is valid, so the three-channel algorithm can be used, if

float{*M(ir37, n; 0, 0, k, cell)*} / float{*M(ir11, n; 0, 0, k, cell)*} ≥ *IR37_THRESH*.

Otherwise use the two-channel algorithm. The two-channel algorithm is always used for day-time data, that is, if *nadir_day(k, cell)* = TRUE.

Step 4.9.2.3

Calculate the averaged SST using the nadir-view retrieval coefficients for the appropriate across-track band given by *j* = *map*(*across_track_mean(0; k, cell)*) and for the two or three channel algorithm as appropriate, for each latitude zone i = 0, 1, 2:

Step 4.9.2.3.1

Perform this step if the 3.7 micron channel is not available for use.

The equations for use with the nadir view are

$$T_{sst,i}^{nadir} = 100.0a_0 + a_1T_{ir11}^{nadir} + a_2T_{ir12}^{nadir}$$

where

$$a_q = a(i, j, q).$$

(Req 4.9-3)

(Here and elsewhere in this module the factor of 100 is to ensure consistency of units between the brightness temperatures, in units of 0.01K, and the leading coefficient, in K.)

Set

nadir_asst_uses_ir37(k, cell) = FALSE

(Req 4.9-3.1)

Step 4.9.2.3.2

Perform this step IF the 3.7 micron channel is to be used.

The equations for use with the nadir view are

$$T_{sst,i}^{nadir} = 100.0b_0 + b_1T_{ir11}^{nadir} + b_2T_{ir12}^{nadir} + b_3T_{ir37}^{nadir}$$

where

$$b_q = b(i, j, q).$$

(Req 4.9-4)

(As before, the factor of 100 is to ensure consistency of units.)

Set

nadir_asst_uses_ir37(k, cell) = TRUE

(Req 4.9-4.1)

Step 4.9.2.4

Return latitude-corrected SST (with linear interpolation).

If the cell is in the polar or tropical zone, return the corresponding retrieval. If the cell is in the temperate zone, use linear interpolation with respect to latitude to derive the averaged SST from the values for the temperate zone and the appropriate adjacent zone.

IF abs(*latitude*) < TROPICAL_INDEX then

$$T\_nadir(k,cell) = T_{sst,0}^{nadir}$$

(Req 4.9-5)

IF TROPICAL_INDEX ≤ abs(*latitude*) < TEMPERATE_INDEX, the final value for the retrieved sst is given by

$$T\_nadir(k,cell) = T_{sst,0}^{nadir} + w\cdot(T_{sst,1}^{nadir} - T_{sst,0}^{nadir})$$

(Req 4.9-6)

where

$$w = \frac{(\text{abs}(latitude) - TROPICAL\_INDEX)}{(TEMPERATE\_INDEX - TROPICAL\_INDEX)}$$

IF the TEMPERATE_INDEX ≤ abs(*latitude*) < POLAR_INDEX the but NOT LESS than, the final value for the retrieved sst is given by

$$T\_nadir(k,cell) = T_{sst,1}^{nadir} + w\cdot(T_{sst,2}^{nadir} - T_{sst,1}^{nadir})$$

(Req 4.9-7)

where

$$w = \frac{(\text{abs}(latitude) - TEMPERATE\_INDEX)}{(POLAR\_INDEX - TEMPERATE\_INDEX)}$$

If POLAR_INDEX ≤ abs(*latitude*)

$$T\_nadir(k,cell) = T_{sst,2}^{nadir}$$

(Req 4.9-8)

Step 4.9.3 Dual view average.

Calculate the dual view averaged SST value for the 10-arcmin cells.

Step 4.9.3.1

Determine the minimum numbers of pixels required for the cell, for both views, using the latitude value representative of the cell. The latitude dependent threshold for the nadir view is *minpn* calculated as above. That for the forward view is.

$$minpf = 340 * FRWRD\_PIXELS\_THRESH * \cos((\pi/180.)*latitude) + 1.$$

IF the number of valid pixels in the either view is LESS THAN the threshold value calculated, move to the next 10-arcmin cell.

IF

M(ir12, n; 0, 0, *k, cell)* ≥ *minpn* and *M(ir11, n*; 0, 0, *k, cell)* ≥ *minpn*

AND

M(ir12, f; 0, 0, *k, cell)* ≥ *minpf* and *M(ir11, f*, 0, 0, *k, cell)* ≥ *minpf*

proceed to calculate the retreived sst as below, otherwise set

T_dual(k, cell) = -1.

(Req 4.9-9)

dual_asst_uses_ir37(*k, cell*) = *FALSE*

(Req 4-9-9a)

Step 4.9.3.2

For night-time data, defined by the condition
(*nadir_day(k, cell)* = FALSE and *frwrd_day(k, cell)* = FALSE),
test whether the ratio of pixels with valid 3.7 µm data in each view is greater or less than the threshold value and use the appropriate (two or three channel) SST algorithm. The 3.7 micron channel is valid if

float{*M(ir37, n; 0, 0, k, cell)* + *M(ir37, f; 0, 0, k, cell)*} /
float{*M(ir11, n; 0, 0, k, cell)* + *M(ir11, f; 0, 0, k, cell)*} ≥ *IR37_THRESH*.

Otherwise use the two-channel algorithm. The two-channel algorithm is always used for day-time data, defined by the condition
(*nadir_day(k, cell)* = TRUE or *frwrd_day(k, cell)* = TRUE).

(Req 4.9-10)

Step 4.9.3.3

Calculate the averaged SST using the dual-view retrieval coefficients for the appropriate across-track band given by j = *map*(*across_track_mean(0; k, cell)*) and for the two or three channel algorithm as appropriate, for each latitude zone.

(Req 4.9-11)

Step 4.9.3.3.1

Perform this step if the 3.7 micron channel is not available for use.

The algorithm using both views is given by

$$T_{sst,i}^{dual} = 100.0c_0 + c_1 T_{ir11}^{nadir} + c_2 T_{ir12}^{nadir} + c_3 T_{ir11}^{frwrd} + c_4 T_{ir12}^{frwrd}$$

(Req 4.9-12)

where

$$c_q = c(i, j, q).$$

Set

dual_asst_uses_ir37(k, cell) = *FALSE*

(Req 4.9-12.1)

Step 4.9.3.3.2

Perform this step if the 3.7 micron channel is to be used.

$$T_{sst,i}^{dual} = 100.0d_0 + d_1 T_{ir11}^{nadir} + d_2 T_{ir12}^{nadir} + d_3 T_{ir37}^{nadir} + d_4 T_{ir11}^{frwrd} + d_5 T_{ir12}^{frwrd} + d_6 T_{ir37}^{frwrd}$$

(Req 4.9-13)

where

$$d_q = d(i, j, q).$$

Set

dual_asst_uses_ir37(k, cell) = TRUE

(Req 4.9-13.1)

Step 4.9.3.4

Return latitude-corrected SST (with linear interpolation).

IF abs(*latitude*) < TROPICAL_INDEX THEN

$$T\_dual(k, cell) = T_{sst,0}^{dual}$$

(Req 4.9-14)

IF TROPICAL_INDEX ≤ abs(*latitude*) < TEMPERATE_INDEX, the final value for the retrieved sst is given by

$$T\_dual(k, cell) = T_{sst,0}^{dual} + w \cdot (T_{sst,1}^{dual} - T_{sst,0}^{dual})$$

(Req 4.9-15)

where

$$w = \frac{(\mathrm{abs}(latitude) - TROPICAL\_INDEX)}{(TEMPERATE\_INDEX - TROPICAL\_INDEX)}$$

IF TEMPERATE_INDEX ≤ abs(latitude) < POLAR_INDEX the final value for the retrieved sst is given by

$$T\_dual(k, cell) = T_{sst,1}^{dual} + w \cdot (T_{sst,2}^{dual} - T_{sst,1}^{dual})$$

(Req 4.9-16)

where

$$w = \frac{(\mathrm{abs}(latitude) - TEMPERATE\_INDEX)}{(POLAR\_INDEX - TEMPERATE\_INDEX)}$$

IF POLAR_INDEX ≤ abs(*latitude*)

$$T\_dual(k, cell) = T_{sst,2}^{dual}$$ (Req 4.9-17)

Step 4.9.4

For up to nine 10-arcmin cells within the half-degree cell, derive the mean nadir view SST for the half-degree cell, and the standard deviation of the 10-arcmin SST values. Repeat for the dual-view retrieval. That is

$$T\_nadir(cell) = \frac{1}{\mu_1}\sum_k T\_nadir(k,cell)$$

$$T\_dual(cell) = \frac{1}{\mu_2}\sum_k T\_dual(k,cell)$$

(Req 4.9-18)

where in each case the sum is over all values of k for which the respective sub-cell temperature is valid (i.e. has a positive value), and where μ_1 and μ_2 are the numbers of such valid temperatures in the nadir and forward views, respectively. If either of the values μ_1 or μ_2 is zero, set the corresponding temperature to –1.

The mean across-track pixel number to be associated with the 30 arc minute SST is [L2-INT-369]

$$sst\_mean\_pixel(0;\ cell) = \frac{1}{\mu_1}\sum_k across\_track\_mean(0;k,cell) \text{ if } \mu_1 > 0$$

$$sst\_mean\_pixel(0;\ cell) = -1 \text{ if } \mu_1 = 0$$

where the sum is over all $k \in \{0 \le k \le 8\}$ for which corresponding SST *T_nadir*(*k*, *cell*) is valid (not equal to –1.)

Step 4.9.5 Prepare the confidence flag words for the cell.

The confidence flag word for the sub-cell indexed by (*k*, *cell*) should be prepared as follows:

Set bit 0 if 3-channel algorithm was used at Step 4.9.2.2.2, i.e. if *nadir_asst_uses_ir37(k, cell)* = *TRUE*, otherwise clear bit.

Set bit 1 if 3-channel algorithm was used at Step 4.9.2.3.2, i.e. if *dual_asst_uses_ir37(k, cell)* = *TRUE*, otherwise clear bit.

Set bit 2 if *nadir_day(k, cell)* from §4.8.3 is TRUE, otherwise clear bit.

Set bit 3 if *frwrd_day(k, cell)* from §4.8.3 is TRUE, otherwise clear bit.

The confidence flag word for the half-degree cell indexed by *cell* - to go in AST MDS#1 - will be derived by ORing together the words for those sub-cells (*k*, *cell*), *k* = 0, ...8, for which a valid temperature was derived.

(Req 4.9-19)

4.10 Module Definition: Averaged LST and NDVI Retrieval (Half Degree Cell)

4.10.1 Functional Description

The Land Surface Temperature (LST) and Normalised Difference Vegetation Index (NDVI) are calculated for each sub-cell for which average reflectances over land have been calculated. The averaged LST and NDVI over all the subcells, and their standard deviation, are also computed.

4.10.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-AUX3-14		NADIR_PIXELS_THRESH	float	none	4	1
L2-AUX5-1		Coefficient A0 (day-time) for LST	float	K	4	1
L2-AUX5-2		Coefficient A1 (day-time) for LST	float	none	4	1
L2-AUX5-3		Coefficient A2 (day-time) for LST	float	none	4	1
L2-AUX5-4		Coefficient A0 (night-time) for LST	float	K	4	1
L2-AUX5-5		Coefficient A1 (night-time) for LST	float	none	4	1
L2-AUX5-6		Coefficient A2 (night-time) for LST	float	none	4	1
L2-AUX6-1		Vegetation class index [360][720] for LST	ss	n/a	2	720
L2-AUX7-1		Vegetation fraction[12][360][720]	ss	0.001	2	720
L2-AUX8-1		Precipitable water[12][360][720]	ss	0.01 mm	2	720
L2-AUX9-1		Topographic Variance Flag[360][720]	ss	n/a	2	720
L2-AUX10-1	d	Water vapour factor for LST retrieval	float	none	4	1
L2-AUX10-2	m	Angle factor for LST retrieval	ss	none	2	1
L2-AUX10-3	N_CLASS	Number of vegetation classes for LST	ss	none	2	1

Table 4.10-1: Input Data Table - LST Retrieval LUTs and auxiliary parameters

Parameter ID	Variable	Name	Type	Units	Field size	Fields
	k	sub-cell number k = 1, ..., 9	sl	none	4	
	cell	cell identifier	sl	none	4	
		sub-cell total ch = 1, ..., 7	sl	n/a	4	
L2-INT-38	A(v870, n; 1, 0, k, cell)	sub-cell reflectance average, 0.870 micron channel, nadir view, clear, land	ss	0.01%	2	
L2-INT-37	M(v870, n; 1, 0, k, cell)	sub-cell pixel count, 0.870 micron channel, nadir view, clear, land	ss	none	2	
L2-INT-38	A(v670, n; 1, 0, k, cell)	sub-cell reflectance average, 0.670 micron channel, nadir view, clear, land	ss	0.01%	2	
L2-INT-37	M(v670, n; 1, 0, k, cell)	sub-cell pixel count, 0.670 micron channel, nadir view, clear, land	ss	none	2	
L2-INT-90	NDVI(k, cell)	Averaged NDVI in 10-arcmin cells	ss	0.0001	2	k = 0, 8
L2-INT-91	<NDVI>(cell)	mean NDVI	ss	0.0001	2	per cell
L2-INT-92	σ(NDVI; cell)	standard deviation of NDVI	ss	0.0001	2	per cell
local	μ	number of sub-cells contributing to cell mean	sl	none	4	per cell
L2-INT-93	N0(cell)	Number of pixels in NDVI average, half degree cell	us	none	2	per cell
L2-INT-94	N1(k, cell)	Number of pixels in NDVI average, 10 arc min cells	us	none	2	k = 0, 8
L2-INT-364	ast_conf(1; k, cell)	AST confidence word for land sub-cell	sl	flags	4	k = 0, 8
L2-INT-365	ast_conf(1; cell)	AST confidence word for land cell	sl	flags	4	per cell
	The following parameters are required by the Land Surface Temperature algorithm:					
L2-INT-26	time(sg)	scan UTC	double	days	8	per sg
L2-INT-47	cell_lat(cell)	cell latitude	sl	μdeg	4	per cell
L2-INT-48	cell_long(cell)	cell longitude	sl	μdeg	4	per cell
L2-INT-470		vegetation_class(lat_index, lon_index)	ss	none	2	360 × 720
L2-INT-471		vegetation_fraction(lat_index, lon_index)	ss	0.001	4	360 × 720
L2-INT-472		precipitable_water(lat_index, lon_index)	ss	0.01 mm	2	360 × 720
L2-INT-473		Topographic_flag(lat_index, lon_index)	ss	none	2	360 × 720
L2-INT-474	lat_index	Latitude index: lat_index = 0, …359	sl	none	4	1
L2-INT-475	lon_index	Longitude index: lon_index = 0, 719	sl	none	4	1
L2-INT-489	disp_lat_index	Displaced latitude index: = 0, …359	sl	none	4	1
L2-INT-490	disp_lon_index	Displaced longitude index: = 0, 719	sl	none	4	1
L2-INT-476	month	Index to month: month = 0, 11	sl	none	4	1

L2-INT-477	sun_elev	Solar elevation at land pixel	float	degrees	4	1
L2-INT-478	sat_elev	Satellite elevation at land pixel	float	degrees	4	1
L2-INT-479	night	Day/night flag	sl	none	2	1
L2-INT-480	n	Non-linear exponent	float	none	4	1
L2-INT-481	f	Vegetation fraction at pixel	float	none	4	1
L2-INT-482	pw	Precipitable water at pixel	float	cm	4	1
L2-INT-482	coeff(class, i, j, 0)	Sub-array of coefficients A0	float	0.01K	4	64
L2-INT-484	coeff(class, i, j, 1)	Sub-array of coefficients A1	float	none	4	64
L2-INT-485	coeff(class, i, j. 2)	Sub-array of coefficients A2	float	none	4	64
L2-INT-486	w	Interpolation weight	float	none	4	1
L2-INT-487	a(k)	Retrieval coefficients for pixel	float	mixed	4	1
Local	lst	Land surface temperature at pixel	float	0.01K	4	1
Local	pw00	Precipitable water sample value	ss	0.01 mm	2	1
Local	pw01	Precipitable water sample value	ss	0.01 mm	2	1
Local	pw10	Precipitable water sample value	ss	0.01 mm	2	1
Local	pw11	Precipitable water sample value	ss	0.01 mm	2	1
Local	q	Latitude argument for bilinear interpolation	float	none	4	1
Local	p	Longitude argument for bilinear interpolation	float	none	4	1
Local	class	Index to table of cefficients	sl	none	4	1
Local	latitude	Temporary latitude value	float	Degrees	4	1
Local	minpn	Minimum nadir pixels	float	none	4	1
Local	kx	Sub-cell index in longitude	sl	none	4	1
Local	ky	Sub-cell index in latitude	sl	none	4	1
L2-INT-32	sub_cell_lat(k, cell)	sub-cell latitude	sl	μdeg	4	k = 0, 8
L2-INT-49	nadir_day(k, cell)	nadir view day/night flag	ss	flag	2	k = 0, 8
L2-INT-50	frwrd_day(k, cell)	forward view day/night flag	ss	flag	2	k = 0, 8
L2-INT-60	band(j)	number of across track band (or strip)	sl	none	4	j = 0, 511
L2-INT-121		nadir_band_edge_satellite_elevation(i, k)	float	degrees	4	k = 0, 10
L2-INT-359	across_track_mean(sf; k, cell)	mean across-track pixel index, subcell k	ss	none	2	k = 0, 8
L2-INT-491	sub_cell_index(k, cell)	Along-track index representative of sub-cell	sl	none	4	k = 0, 8
L2-INT-492	T_land(k, cell)	Land surface temperature in sub-cell k.	ss	0.01 K	4	k = 0, 8
L2-INT-493	T_land(cell)	Averaged land surface temperature in cell.	ss	0.01 K	4	per cell
L2-INT-494	σ_land(cell)	Standard deviation of Averaged LST	ss	0.01 K	4	per cell
L2-INT-364	ast_conf(sf; k, cell)	AST confidence word for sub-cell	sl	flags	4	k = 0, 8
L2-INT-365	ast_conf(sf; cell)	AST confidence word for cell	sl	flags	4	per cell
L2-INT-369	sst_mean_pixel(sf; cell)	Mean across-track pixel index, cell	ss	none	2	per cell

Table 4.10-2: Internal Data Table - Averaged LST and NDVI Retrieval (Half Degree Cell)

4.10.3 Detailed Structure

The following processing is applied to cells when the processing of Step 4.8.1 is complete; i.e. no more pixels remain to be added to the cell.

Step 4.10.1 Calculate subcell NDVIs.

NDVI is defined by

$$NDVI(k, cell) = 10000 \frac{A(v870, n;1,0, k, cell) - A(v670, n;1,0, k, cell)}{A(v870, n;1,0, k, cell) + A(v670, n;1,0, k, cell)}$$

(Req 4.10-1)

provided both values are valid (not exceptional). Otherwise set

$NDVI(k, cell) = -19999.$

The number of pixels contributing to the sub-cell mean, *N*1(*k, cell*), provided as a confidence indicator, is the smaller of *M(v870, n; 1, 0, k, cell)* and *M(v670, n; 1, 0, k, cell)*.

Step 4.10.2 Calculate cell NDVI.

The mean in the larger (30 arc minute) cell is given by

$$\langle NDVI\rangle(cell) = \frac{1}{\mu}\sum_k NDVI(k, cell) \qquad \text{(Req 4.10-2)}$$

where the sum is over all $k \in \{0 \le k \le 8\}$ having a valid subcell mean *NDVI* and μ is the number of such valid means. The number of pixels that contribute to the mean is similarly the smaller of

$$\tilde{M}(v870, n;1,0, cell),\ \tilde{M}(v670, n;1,0, cell).$$

The standard deviation of the mean is

$$\sigma(NDVI; cell) = \left\{\frac{1}{\mu-1}\sum_k (NDVI(k, cell) - \langle NDVI\rangle(cell))^2\right\}^{1/2} \qquad \text{(Req 4.10-3)}$$

in all cases the sum is over sub-cells having valid means.

If the number of valid subcell means μ is zero, set

<NDVI>(*cell*) = -19999.

(Req 4.10-4)

If the number of valid subcell means μ ≤ 1, so that a valid standard deviation cannot be calculated, set set

$$\sigma(NDVI; cell) = -19999.$$

(Req 4.10-4.1)

Step 4.10.3 Read in coefficients and auxiliary tables for LST retrieval.

The coefficients for LST retrieval are identical to those used for the full resolution product, as read in by Step 4.6.1.2 (Section 4.6.3). If these coefficients are still available in the processor, there is no need to repeat the following steps.

Step 4.10.3.1 Read in coefficients

For each of the *N_CLASS* vegetation classes there are two records, for vegetation and for bare soil. Open the file of retrieval coefficients L2-AUX5.

The LST coefficient set is read in as follows.

for *class* = 0, *N_CLASS* - 1 (outer loop)
  for *i* = 0, 1 (inner loop)
    coeff(class, i, 0, 0) = [L2-AUX5-1]
    coeff(class, i, 0, 1) = [L2-AUX5-2]
    coeff(class, i, 0, 2) = [L2-AUX5-3]
    coeff(class, i, 1, 0) = [L2-AUX5-4]

coeff(class, i, 1, 1) = [L2-AUX5-5]
coeff(class, i, 1, 2) = [L2-AUX5-6]

(Req 4.10-5)

Step 4.10.3.2 Determine month index

Using a suitable calendar function, determine the month (*month* = 0, … 11) in which the data were collected from the scan time of start of data *time*(0)=[L2-INT-26](0):

month = *month*(*time*(0))

(Req 4.10-6)

Step 4.10.3.3 Read in auxiliary files

Note that in the cases of data sets L2-AUX7 and L2-AUX8 only one plane of data, that corresponding to the current month, is required in memory for a given run of the processor.

Read in Vegetation Class Index: Open the vegetation class file L2-AUX6.

for each latitude index *i* = 0, 359
vegetation_class(*i*, *j*) = [L2-AUX6-1](*j*) for all *j* of record *i*.

(Req 4.10-7)

Read in Vegetation Fraction Table: Open the file of vegetation fraction data L2-AUX7.

for each latitude index *i* = 0, 359
select record (360 × *month* + *i*)
vegetation_fraction(*i*, *j*) = [L2-AUX7-1](*j*) for all *j* of selected record.

(Req 4.10-8)

Read in Precipitable Water Data: Open the file of precipitable water data L2-AUX8.

for each latitude index *i* = 0, 359
select record (360 × *month* + *i*)
precipitable_water(*i*, *j*) = [L2-AUX8-1](*j*) for all *j* of selected record.

(Req 4.10-9)

Read in Topographic Variance Flag: Open the file of topographic variance flags L2-AUX9.

for each latitude index *i* = 0, 359
topographic_flag(*i*, *j*) = [L2-AUX9-1](*j*) for all *j* of record *i*.

(Req 4.10-10)

Step 4.10.4 Derive Land Surface Temperature for sub-cells

LST retrieval uses the nadir view 11 and 12 micron channels in conjunction with retrieval coefficients derived from the tables.

Note that as in Section 4.9 we adopt an abbreviated notation for the average brightness temperatures in this section.

$$T_{ch,sf}^{nadir} = \left[\frac{\text{float}(S(ch,n;sf,0,k,cell))}{\text{float}(M(ch,n;sf,0,k,cell))} \right]$$

where *ch* indicates one of the long-wavelength infra-red channels, and where *sf* is the surface type flag. This notation is slightly more complex than that used in Section 4.9 because it is necessary to distinguish between land and sea averages. Where this notation it is used, a dependence on *k* and *cell* is implied. As in Section 4.9, these quantities must be computed using a floating point computation, although *S* and *M* are of type integer, to ensure that sufficient precision is maintained.

The calculation proceeds as follows for each cell in turn.

Step 4.10.4.1 Determine latitude and longitude indices

lat_index = [*cell_lat*(*cell*)/500000] + 180

lon_index = [*cell_lon*(*cell*)/500000] + 360

(Req 4.10-11)

Note: Because *cell_lat* and *cell_long* are defined (Step 4.8.1.1) as integer multiples of 500000, the above integer divisions should be exact, and the values of *lat_index* and *lon_index* should equal the values of *cell_ latitude_index* and *cell_ longitude_index* respectively that were computed locally for the same cell in Step 4.8.1.1.

Extract the vegetation class for the cell:

class= *vegetation_class*(*lat_index*, *lon_index*)

(Req 4.10-12)

Step 4.10.4.2 Test for valid data

For each sub-cell *k* = 0, … 8, if either the 11 or 12 micron brightness temperature in the nadir view is invalid, the calculation is abandoned, and the LST is set to -1. The criterion for invalid data is the same as that used for the SST processing, as follows.

Determine the minimum number of pixels required for the cell, for the nadir view, using the latitude of the cell. This is (in degrees)

$$latitude = sub\_cell\_lat(k, cell) * 10^{-6}$$

The latitude dependent threshold is

$$minpn = 340 * NADIR\_PIXELS\_THRESH * \cos((\pi/180.)*latitude) + 1.$$

(Req 4.10-13)

Identify whether the land or ‘sea’ brightness temperatures are required. Inland lakes are flagged as sea in the current land/sea data-base, so must be treated accordingly.

If *class* = 14 then *sf* = 0 (sea) otherwise *sf* = 1 (land).

If $M(ir12, n; sf, 0, k, cell) \geq minpn$ and $M(ir11, n; sf, 0, k, cell) \geq minpn$ proceed to calculate the retrieved LST as below, otherwise set

$$T\_land(k, cell) = -1.0$$

(Req 4.10-14)

If the 11 and 12 micron nadir brightness temperatures are valid, Steps 4.10.4.3 to 4.10.4.5 are to be repeated for each sub-cell *k* = 0, … 8 in the cell.

Step 4.10.4.3 Determine day/night flag, satellite elevation and non-linear exponent

If *nadir_day(k, cell)* = TRUE then
night = 0 else *night* = 1

(Req 4.10-15)

A linear interpolation is used to determine the satellite elevation:

$j = across\_track\_mean(1, k, cell)$

$w = float(j - 6)/50.0 - band(j)$

$i = sub\_cell\_index(k, cell)$

$sat\_elev = (1.0 - w) \times nadir\_band\_edge\_satellite\_elevation(i, band(j)) + w \times nadir\_band\_edge\_satellite\_elevation(i, band(j) + 1)$

(Req 4.10-16)

Calculate the non-linear exponent:

$\boldsymbol{n} = 1.0 / \cos(\pi \times (90 - sat\_elev) / (m \times 180.0)$

(Req 4.10-17)

Note that m is [L2-AUX10-2] and $\boldsymbol{n}$ is [L2-INT-480].

Step 4.10.4.4 Determine coefficients

$f = 0.001 \times vegetation\_fraction(lat\_index, lon\_index)$

(Req 4.10-18)

ky = integer part of $[k/3]$

$kx = k - 3*ky$

(Req 4.10-19)

If $ky = 0$ and $lat\_index > 0$ then $disp\_lat\_index = lat\_index - 1$
else $disp\_lat\_index = lat\_index$

If $kx = 0$ then $disp\_lon\_index = [720 + lon\_index - 1]$(modulo 720)
else $disp\_lon\_index = lon\_index$

(Req 4.10-20)

Interpolate precipitable water:

$pw00 = precipitable\_water(disp\_lat\_index, disp\_lon\_index)$

$pw01 = precipitable\_water(disp\_lat\_index{+}1, disp\_lon\_index)$

$pw10 = precipitable\_water(disp\_lat\_index, [disp\_lon\_index{+}1]\text{(modulo 720)})$

$pw11 = precipitable\_water(disp\_lat\_index{+}1, [disp\_lon\_index{+}1]\text{(modulo 720)})$

(Req 4.10-21)

If $ky = 0$ then $q = (2.0 / 3.0)$ else $q = (ky - 1) / 3.0$

If $kx = 0$ then $p = (2.0 / 3.0)$ else $p = (kx - 1) / 3.0$

$pw = 0.001 \times ((1 - p)(1 - q)pw00 + (1 - p)q \times pw01 + p(1 - q)pw10 + pq \times pw11)$

(Req 4.10-22)

$class = vegetation\_class(lat\_index, lon\_index) - 1$

(Req 4.10-23)

If *class* < 0 or *class* > *NCLASS* - 1 then the index is out of range; the calculation for this sub-cell is abandoned and the nadir field should be set to an exception value of -1.0:

T_land(k, cell) = -1.0

(Req 4.10-24)

Otherwise

for $k = 0, 2$

$$a(k) = f \times coeff(class, 0, night, k) + (1.0 - f) \times coeff(class, 1, night, k)$$

(Req 4.10-25)

If (*class* + 1) = 14 this cell is flagged as an inland lake in the vegetation class database. The exponent *n* and the precipitable water correction are not used, and the correct brightness temperature average to be used is that for pixels flagged as sea. Set

n = 1.0.

Otherwise if (*class* + 1) ≠ 14 correct *a*(0) as follows:

$$a(0) = a(0) + d \times (cosec(\pi \times sat\_elev / 180.0) - 1.0) \times pw$$

(Req 4.10-26)

Note that *d* is [L2-AUX10-1].

Step 4.10.4.5 Calculate the land surface temperature.

Note that the surface flag index *sf* retains the value assigned in Step 4.10.4.2.

If $T^{nadir}_{ir11,sf} > T^{nadir}_{ir12,sf}$ then

$$lst = 100. \times (a(0) + a(1) \times (0.01 \times (T^{nadir}_{ir11,sf} - T^{nadir}_{ir12,sf})))^{**}\boldsymbol{n}) + (a(1) + a(2)) \times (T^{nadir}_{ir12,sf} - 27315) + 27315$$

(Req 4.10-27)

else

$$lst = 100. \times a(0) + a(1) \times (T^{nadir}_{ir11,sf} - T^{nadir}_{ir12,sf}) + (a(1) + a(2)) \times (T^{nadir}_{ir12,sf} - 27315) + 27315$$

(Req 4.10-28)

Set appropriate bits on AST confidence word *ast_conf*(1; *k*, *cell*):

Set bit 2 if *nadir_day(k, cell)* from §4.8.3 is TRUE, otherwise clear bit.

(Req 4.10-29)

Set bit 3 if *frwrd_day(k, cell)* from §4.8.3 is TRUE, otherwise clear bit.

(Req 4.10-30)

Set bits 4 and 5 to the topographic variance flags:

[*ast_conf*(1; *k*, *cell*)](bits 4:5) = *topographic_flag*(*lat_index*, *lon_index*)

(Note that this is a two-bit flag.)

```
                                                      (Req 4.10-31)
```

Trap for LST out of range:

If *lst* ≥ 32767.5 then

```
    T_land(k, cell) = -1                              (Req 4.10-32)
```

else

```
    T_land(k, cell) = integer part of (lst + 0.5)     (Req 4.10-33)
```

Step 4.10.5 Calculate 30 arc min average

For up to nine 10-arcmin cells within the half-degree cell, derive the mean LST [L2-INT-493] for the half-degree cell, and the standard deviation [L2-INT-494] of the 10-arcmin LST values.

$$T\_land(cell) = \frac{1}{\mu}\sum_k T\_land(k, cell)$$

`(Req 4.10-34)`

where the sum is over all values of *k* for which the respective sub-cell LST is valid (i.e. has a positive value), and where μ is the number of valid temperatures. If μ is zero, set the corresponding temperature to –1. To calculate the standard deviation [L2-INT-494] use an expression analogous to Req 4.10-3.

The mean across-track pixel number to be associated with the 30 arc minute LST is [L2-INT-369]

$$sst\_mean\_pixel(1; cell) = \frac{1}{\mu}\sum_k across\_track\_mean(1; k, cell) \text{ if } \mu > 0$$

$$sst\_mean\_pixel(1; cell) = -1 \text{ if } \mu = 0$$

`(Req 4.10-35)`

where the sum is over all $k \in \{0 \le k \le 8\}$ for which corresponding LST *T_land*(*k*, *cell*) is valid (not equal to –1).

Derive the confidence flag word *ast_conf*(1; *cell*) for the half-degree cell indexed by *cell* by ORing together the words for those sub-cells (*k*, *cell*), *k* = 0, ...8, for which a valid temperature was derived. (Note that the topographic variance bits will be the same for all sub-cells in the cell.)

`(Req 4.10-36)`

4.11 Module Definition: Spatially Averaged Cloud Parameters (Half Degree Cell)

4.11.1 Functional Description

This module is to provide physical information on the cloud state additional to the results of the cloud flagging provided by the cloud clearing algorithms. The product is based on the same half-degree cells defined above. The frequency distribution of brightness temperature for the cloudy pixels within the cell is given together with representative parameters and an estimate of the cloud-top temperature. The latter is interpreted as the mean brightness temperature of the coldest 25% of the cloudy pixels in the cell.

For each half-degree cell, information is given for the nadir and forward views separately. The information consists of the number of cloudy and cloud-free pixels falling within the cell, a histogram of the 11 micron brightness temperatures of the cloudy pixels, and various statistical parameters derived from the histogram. The 11 micron channel is used as the basis of the product following the practice of ATSR and ATSR-2.

The product is generated as follows. Two histograms are generated of the frequency distribution of 11 micron brightness temperature, for cloudy pixels over sea and land respectively. The histograms represent the brightness temperature at 0.1 K resolution between 190 K and 290 K. Thus each contains 1000 bins where the first bin contains the number of pixels with temperatures in the range 190.0 to 190.1 K, and the last bin contains the number of pixels with temperatures in the range 289.9 to 290.0 K. The cloud state of each filled pixel falling within the cell is inspected. If it is clear, a count of the number of clear pixels is incremented; if it is cloudy, the 11 micron channel BT is inspected and the count in the appropriate histogram bin is incremented. Note that cosmetic fill pixels are included in the processing.

As each pixel is inspected, a test is made to determine whether its 11 micron BT is lower than the lowest value previously encountered, and if so to store the location of the pixel. Then when the histogram is complete the identity of the minimum pixel will be known, and can be used to extract its channel values.

Once the histogram is complete for a given cell, that is once all the pixels falling within the cell have been inspected, the cloud temperature and coverage results are derived from it. Firstly the total number of cloudy pixels detected is computed by summing the histogram samples. If this total is less than 20 no further derivations are performed. If 20 or more cloudy pixels have been identified and included in the histogram, the mean 11 micron brightness temperature and its standard deviation are calculated from the histogram.

The histogram is searched for the lowest temperature represented by the histogram. This is the temperature corresponding to the first non-zero bin of the histogram. Next, the cloud-top temperature is estimated. The histogram bin containing the 25th percentile is identified; this is the first bin (as the histogram is searched in the direction of ascending temperature) for which the cumulative total of the bins up to and including itself exceeds 25% of the total number of cloudy pixels. The mean temperature represented by the bins up to and including this bin is calculated.

[Note that the cloud top temperature so derived may represent the mean of slightly more than 25% of the cloudy pixels, since the cumulative total including the 25th percentile bin may exceed 25%.]

Finally the percentage cloud cover is calculated from the ratio of cloudy pixels to total pixels.

4.11.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields

L2-INT-101	I(ir12, n; i, j)	regridded nadir ir12 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-102	I(ir11, n; i, j)	regridded nadir ir11 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-103	I(ir37, n; i, j)	regridded nadir ir37 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-104	I(v16, n; i, j)	regridded nadir v16 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-105	I(v870, n; i, j)	regridded nadir v870 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-106	I(v670, n; i, j)	regridded nadir v670 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-107	I(v555, n; i, j)	regridded nadir v555 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-111	I(ir12, f; i, j)	regridded forward ir12 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-112	I(ir11, f; i, j)	regridded forward ir11 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-113	I(ir37, f; i, j)	regridded forward ir37 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-114	I(v16, f; i, j)	regridded forward v16 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-115	I(v870, f; i, j)	regridded forward v870 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-116	I(v670, f; i, j)	regridded forward v670 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-117	I(v555, f; i, j)	regridded forward v555 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-100	nadir_fill_state(i, j)	nadir fill state indicator	byte	none	1	j = 0, 511
L2-INT-110	frwd_fill_state(i, j)	frwd fill state indicator	byte	none	1	j = 0, 511
L2-INT-232	nadir_land(i, j)	nadir land/sea flag	ss array	flag	2	j = 0, 511
L2-INT-233	nadir_cloud(i, j)	nadir cloud state flag	ss array	flag	2	j = 0, 511
L2-INT-248	frwd_land(i, j)	forward land/sea flag	ss array	flag	2	j = 0, 511
L2-INT-249	frwd_cloud(i, j)	forward cloud state flag	ss array	flag	2	j = 0, 511
L2-INT-160	image_lat(i, j)	image pixel latitude	float	degrees	4	
L2-INT-161	image long(i, j)	image pixel longitude	float	degrees	4	
local	k	histogram bin counter k = 0, 999	sl	none	4	
local	K	25th percentile count	ss	none	2	
	cell	cell number	sl	none	4	
	I_min(ch, v, sf; cell)	Brightness temperature or reflectance, as appropriate, of channel ch, view v, for cloudy pixel having minimum 11 micron BT over surface type sf.	ss	0.01 K or 0.01%	2	
L2-INT-325	across_track_band	across-track band	ss	none	2	per cell
		land pixel histogram quantities				
L2-INT-328	nadir_clear_land	total of clear land pixels, nadir view	sl	none	4	per cell
L2-INT-329	frwd_clear_land	total of clear land pixels, forward view	sl	none	4	per cell
L2-INT-330	nadir_cloudy_land	total of cloudy land pixels, nadir view	sl	none	4	per cell
L2-INT-331	frwd_cloudy_land	total of cloudy land pixels, forward view	sl	none	4	per cell
L2-INT-332	nadir_hist_land(k)	nadir histogram (land cell)	ss	none	2	1000
L2-INT-333	frwd_hist_land(k)	forward histogram (land cell)	ss	none	2	1000
		nadir output quantities				
L2-INT-335	bt_cloud_top	cloud top temperatuere (over land)	ss	0.01K	2	per cell
L2-INT-336	bt_percent_cloudy	percentage cloudy pixels (over land)	ss	0.01%	2	per cell
		frwd output quantities				
L2-INT-337	bt_cloud_top	cloud top temperatuere (over land)	ss	0.01K	2	per cell
L2-INT-338	bt_percent_cloudy	percentage cloudy pixels (over land)	ss	0.01%	2	per cell
		sea pixel histogram quantities				
L2-INT-344	nadir_clear_sea	total of clear sea pixels, nadir view	sl	none	4	per cell
L2-INT-345	frwd_clear_sea	total of clear sea pixels, forward view	sl	none	4	per cell
L2-INT-346	nadir_cloudy_sea	total of cloudy sea pixels, nadir view	sl	none	4	per cell
L2-INT-347	frwd_cloudy_sea	total of cloudy sea pixels, forward view	sl	none	4	per cell
L2-INT-348	nadir_hist_sea(k)	nadir histogram (sea cell)	ss	none	2	1000
L2-INT-349	frwd_hist_sea(k)	forward histogram (sea cell)	ss	none	2	1000
		nadir output quantities				
L2-INT-351	bt_cloud_top	cloud top temperatuere (over sea)	ss	0.01K	2	per cell
L2-INT-352	bt_percent_cloudy	percentage cloudy pixels (over sea)	ss	0.01%	2	per cell
		forward output quantities				
L2-INT-353	bt_cloud_top	cloud top temperatuere (over sea)	ss	0.01K	2	per cell
L2-INT-354	bt_percent_cloudy	percentage cloudy pixels (over sea)	ss	0.01%	2	per cell

Table 4-11-1: Internal Data Table - Spatially Averaged Cloud Parameters (Half Degree Cell)

4.11.3 Detailed Structure

Step 4.11.1 Derive histogram for each cell.

Each image row *i* and pixel *j* is used as follows.

Step 4.11.1.1 Identify cell number.

Identify the half-degree cell number *cell* within which the pixel indexed by *i* and *j* falls exactly as in Section 4.8.3, Step 4.8.1.1

IF this is the first pixel to fall within a given cell, clear the histogram arrays:

<view>_histogram_<surface>(cell, k) = 0 for all k = 0, 999,

initialise the clear pixel counters

<view>_clear_<surface> = 0,

AND initialise associated variables for each channel *ch*:

I_min(ch, v, sf; cell) = 29000 if *ch* = *ir*12, *ir*11, *ir*37

I_min(ch, v, sf; cell) = 10000 if *ch* = *v*16, v870, *v*670, *v*555

FOR each view <*v*> = *nadir*, *frwrd*, and for each surface type <*sf*> = *land*, *sea*.

(Req 4.11-1)

Step 4.11.1.2 Process the image pixel.

Process the image pixel at i, j for both nadir and forward views, as follows,

(a) IF the pixel is unfilled, do nothing.

(b) IF the pixel is over sea, and is clear, increment <view>_clear_sea.

(c) IF the pixel is over sea, and is cloudy, check the 11 micron brightness temperature. If the 11 micron brightness temperature

$T_{11} = I(ir11, v; i, j)$

is valid, and if $19000 \leq T_{11} < 29000$, increment the element of the histogram array <view>_histogram_sea(cell, k) specified by index

k = integer part of (T_{11}/10 - 1900)

IF T_{11} < *I_min(ir11, v, sf; cell)* THEN set

I_min(ch, v, sf; cell) = I(ch, v; i, j)

for each channel *ch*.

(d) IF the pixel is over land, treat similarly but increment the land counters and histogram arrays <view>_clear_land and <view>_histogram_sea(cell, k) in place of the corresponding sea variables.

(Req 4.11-2)

Step 4.11.2 Process histograms.

In this step the following notation is used.

$N_{v,sf}^{cloud}(cell)$	<view>_cloudy_<surface>
$F_{v,sf}(cell;k)$	<view>_hist_<surface>(cell, k) where <view> = nadir | frwrd and <surface> = sea | land
$T_{v,sf}^{ct}(cell)$	[bt_cloud_top](v, sf)

When all four histograms are complete, find the number of cloudy pixels in each. For each view v = n, f and for each surface type sf:

$$N_{v,sf}^{cloud}(cell) = \sum_{k=0}^{999} F_{v,sf}(cell;k)$$ (Req 4.11-3)

If the number of cloudy pixels found is less than 20, proceed to the next cell. Otherwise proceed as follows:

Calculate the position of the 25th percentile

$$K = N_{v,sf}^{cloud}(cell)/4$$ (Req 4.11-4)

and find the index k(such that

$$\sum_{k=0}^{k'-1} F_{v,sf}(cell;k) < K \le \sum_{k=0}^{k'} F_{v,sf}(cell;k)$$ (Req 4.11-5)

Then the cloud-top temperature is given by

$$T_{v,sf}^{ct}(cell) = 19000 + 10 \cdot \sum_{k=0}^{k'} (k+0.5) \cdot F_{v,sf}(cell;k) \Big/ \sum_{k=0}^{k'} F_{v,sf}(cell;k)$$

(Req 4.11-6)

and the percentage of cloudy pixels for each view and surface type is given by

$$\text{bt\_percent\_cloudy} = 10000 \cdot \frac{N_{v,sf}^{cloud}(cell)}{N_{v,sf}^{cell}(cell) + <view>\_clear\_<surface>(cell)}$$

(Req 4.11-7)

4.12 Module Definition: Spatial Averaging (50 km Cell)

4.12.1 Functional Description

For the averaged products in 50 km cells, the swath is divided into cells nominally 50 km square, and these cells are further subdivided into 9 square sub-cells of nominal dimension 17 km. For each channel, the average brightness temperature (for the infra-red channels) or radiance (for the visible channels) is averaged over all pixels of each type that fall within each sub-cell, to give distributions of a brightness temperature and radiance at 17 km nominal resolution. Averages are performed for the forward and nadir views separately, and a separate average is performed for each surface type (land and sea) and cloud state (clear or cloudy). There are thus 4 averages per channel per view. The mean across-track band number in each cell is also derived, for use by the averaged SST algorithm.

4.12.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-AUX3-9	NGRANULE	Granule Size	sl	none	4	1
L2-AUX3-10	ΔX, ΔY	AST Cell dimension	sl	none	4	1
L2-AUX3-5		Threshold for ABT flag, 17 km cell nadir view	sl	n/a	4	1
L2-AUX3-6		Threshold for ABT flag, 17 km cell forward view	sl	n/a	4	1
L2-AUX3-7		Threshold for ABT flag, 50 km cell nadir view	sl	n/a	4	1
L2-AUX3-8		Threshold for ABT flag, 50 km cell forward view	sl	n/a	4	1
L2-AUX3-18	MAX_CELLS_X	Number of 50 km cells across-track	ss	none	2	1
L2-AUX3-19	MAX_CELLS_Y	Number of 50 km cells along-track	ss	none	2	1
L2-AUX3-20	MX, j_0	Across-track origin of 50 km cells	sl	none	4	1

Table 4.12-1: Input Data Table - Spatial Averaging, 50 km cell.

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-INT-101	I(1, n; i, j)	nadir ir12 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-102	I(2, n; i, j)	nadir ir11 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-103	I(3, n; i, j)	nadir ir37 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-104	I(4, n; i, j)	nadir v16 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-105	I(5, n; i, j)	nadir v870 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-106	I(6, n; i, j)	nadir v670 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-107	I(7, n, i, j)	nadir v555 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-111	I(1, f; i, j)	forward ir12 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-112	I(2, f; i, j)	forward ir11 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-113	I(3, f; i, j)	forward ir37 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-114	I(4, f; i, j)	forward v16 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-115	I(5, f; i, j)	forward v870 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-116	I(6, f; i, j)	forward v670 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-117	I(7, f, i, j)	forward v555 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-100	nadir_fill_state(i, j)	nadir fill state indicator	byte	none	1	j = 0, 511
L2-INT-110	frwrd_fill_state(i, j)	forward fill state indicator	byte	none	1	j = 0, 511
L2-INT-232	nadir_land(i, j)	nadir view land flag	ss array	flag	2	j = 0, 511
L2-INT-233	nadir_cloud(i, j)	nadir view cloud flag	ss array	flag	2	j = 0, 511
L2-INT-248	frwrd_land(i, j)	forward view land flag	ss array	flag	2	j = 0, 511
L2-INT-249	frwrd_cloud(i, j)	forward view cloud flag	ss array	flag	2	j = 0, 511
L2-INT-160	image_latitude(i, j)	image pixel latitude	float	degrees	4	j = 0, 511
L2-INT-161	image_longitude(i, j)	image pixel longitude	float	degrees	4	j = 0, 511
L2-INT-124		nadir_band_centre_solar_elevation(i, k′)	float	degrees	4	k′ = 0, 9
L2-INT-144		frwrd_band_centre_solar_elevation(i, k′)	float	degrees	4	k′ = 0, 9
(local)	cell_latitude	pixel latitude transformed to cell units	float	cell	4	1
(local)	cell_longitude	pixel longitude transformed to cell units	float	cell	4	1
(local)	cell_latitude_index	cell latitude index	sl	cell	4	1
(local)	cell_longitude_index	cell longitude index	sl	cell	4	1
L2-INT-26	time(sg)	scan UTC	double	days	8	per sg
L2-INT-134	scn_nadir(ig, j)	nadir view instrument scan number	us	none	4	j = 0, 511
L2-INT-135	pxl_nadir(ig, j)	nadir view instrument pixel number	us	none	4	j = 0, 511
L2-INT-154	scn_frwrd(ig, j)	forward view instrument scan number	us	none	4	j = 0, 511

L2-INT-155	pxl_frwrd(ig, j)	forward view instrument pixel number	us	none	4	j = 0, 511
L2-INT-60	band(j)	across-track band number	sl	none	4	512
local	i	image scan index	sl	none	4	
local	j	pixel index j = 0, ..., 511	sl	none	4	
local	ch	channel identifier ch = 1, ..., 7	sl	none	4	
	v	view identifier (nadir / forward)	sl	none	4	
	sf	surface type identifier (0 = sea, 1 = land)	sl	none	4	
	cl	cloud state identifier (0 = clear, 1 = cloud)	sl	none	4	
local	s	index to instrument scans	sl	none	4	1
local	sg	index to ADS #4 records	sl	none	4	1
local	k	sub-cell number k = 1, ..., 9	sl	none	4	
local	l	50 km cell index	sl	none	4	
local	m	50 km cell index m = 0, 1, ..., 9	sl	none	4	
L2-INT-20	utc(l, m)	50 km cell UTC	double	days	8	per cell
L2-INT-21	utc(k, l, m)	17 kmsub-cell UTC	double	days	8	k = 0, 8
L2-INT-62	sub_cell_lat(k, l, m)	sub-cell latitude (17 km)	sl	μdeg	4	
L2-INT-63	sub_cell_long(k, l, m)	sub-cell longitude	sl	μdeg	4	
L2-INT-64	sub_cell_band(k, l, m)	sub-cell across-track band	ss	none	2	
L2-INT-75	nadir_solar_el(k, l, m)	nadir solar elevation for sub-cell	float	degrees	4	
L2-INT-76	frwrd_solar_el(k, l, m)	frwrd solar elevation for sub-cell	float	degrees	4	
L2-INT-77	cell_lat(l, m)	cell latitude (50 km)	sl	μdeg	4	
L2-INT-78	cell_long(l, m)	cell longitude (50 km)	sl	μdeg	4	
L2-INT-79	nadir_day(k, l, m)	nadir view sub-cell day/night flag	ss	flag	2	k = 0, 8
L2-INT-80	frwrd_day(k, l, m)	forward view sub-cell day/night flag	ss	flag	2	k = 0, 8
L2-INT-66	S(ch, v; sf, cl, k, l, m)	sub-cell total ch = 1, ..., 7	sl	n/a	4	
L2-INT-67	M(ch, v; sf, cl, k, l, m)	ch = 1, 2, 3, 4, 5, 6, 7	ss	none	2	
L2-INT-68	A(ch, v; sf, cl, k, l, m)	sub-cell brightness temperature average (For infra-red channels ch = 1, 2, 3)	sl	0.001K	4	
L2-INT-69	A(ch, v; sf, cl, k, l, m)	sub-cell reflectance average (For visible channels ch = 4, 5, 6, 7)	ss	0.01%	2	
L2-INT-70	$\tilde{M}$ (ch, v; sf, cl, l, m)	50 km cell pixel count, ch = 1, ..., 7	ss	none	2	
L2-INT-71	$\tilde{A}$ (ch, v; sf, cl, l, m)	50 km cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	sl	0.001K	4	
L2-INT-72	$\tilde{A}$ (ch, v; sf, cl, l, m)	50 km cell reflectance average (for visible channels ch = 4, 5, 6, 7)	ss	0.01%	2	
L2-INT-73	σ(ch, v; sf, cl, l, m)	standard deviation of the 50 km cell average	float	0.001K or 0.01%	4	
L2-INT-455	N(v; sf, cl, k, l, m)	sub-cell filled pixel count	ss	none	2	
L2-INT-456	band_sum(k, l, m)	cumulative across-track band sum	sl	none	4	
L2-INT-457	mean_band(k, l, m)	mean across-track band number	ss	none	2	
L2-INT-458	across_track_sum(sf; k, l, m)	cumulative sum of across-track pixel index	sl	none	4	k = 0, 8
L2-INT-459	across_track_mean(sf; k, l, m)	mean across-track pixel index, subcell k	ss	none	2	k = 0, 8
L2-INT-468	across_track_mean(sf; l, m)	mean across-track pixel index, cell l, m	ss	none	2	per cell
local	μ	number of sub-cells contributing to mean	sl	none	4	
L2-INT-81	PFF(v, sf; l, m)	Pixel threshold failure flags word, 30 arc minute cell	ss	flags	2	
L2-INT-82	PFF(v, sf; k, l, m)	Pixel threshold failure flags word, 10 arc minute sub-cell	ss	flags	2	
L2-INT-401	N_land(n; k, l, m)	total filled pixels over land for subcell	ss	none	2	k = 0, 8
L2-INT-402	N_sea(n; k, l, m)	total of filled pixels over sea for subcell	ss	none	2	k = 0, 8
L2-INT-403	N_total(n; k, l, m)	total of filled pixels for subcell, nadir view	ss	none	2	k = 0, 8
L2-INT-404	pcs(n; k, l, m)	percentage of cloudy pixels over sea	ss	0.01%	2	k = 0, 8
L2-INT-405	pcl(n; k, l, m)	percentage of cloudy pixels over land	ss	0.01%	2	k = 0, 8
L2-INT-406	N_land(n; l, m)	total filled pixels over land for cell	ss	none	2	
L2-INT-407	N_sea(n; l, m)	total of filled pixels over sea for cell	ss	none	2	
L2-INT-408	N_total(n; l, m)	total of filled pixels for cell, nadir view	ss	none	2	

L2-INT-409	pcs(n; l, m)	percentage of cloudy pixels over sea	ss	0.01%	2	
L2-INT-410	pcl(n; l, m)	percentage of cloudy pixels over land	ss	0.01%	2	
L2-INT-411	N_land(f; k, l, m)	total filled pixels over land for sub-cell	ss	none	2	k = 0, 8
L2-INT-412	N_sea(f; k, l, m)	total of filled pixels over sea for subcell	ss	none	2	k = 0, 8
L2-INT-413	N_total(f; k, l, m)	total of filled pixels for subcell, frwrd view	ss	none	2	k = 0, 8
L2-INT-414	pcs(f; k, l, m)	percentage of cloudy pixels over sea	ss	0.01%	2	k = 0, 8
L2-INT-415	pcl(f; k, l, m)	percentage of cloudy pixels over land	ss	0.01%	2	k = 0, 8
L2-INT-416	N_land(f; l, m)	total filled pixels over land for cell	ss	none	2	
L2-INT-417	N_sea(f; l, m)	total of filled pixels over sea for cell	ss	none	2	
L2-INT-418	N_total(f; l, m)	total of filled pixels for cell, frwrd view	ss	none	2	
L2-INT-419	pcs(f; l, m)	percentage of cloudy pixels over sea	ss	0.01%	2	
L2-INT-420	pcl(f; l, m)	percentage of cloudy pixels over land	ss	0.01%	2	
L2-INT-495	sub_cell_index(k, l, m)	Along-track index representative of sub-cell	sl	none	4	k = 0, 8

Table 4.12-2: Internal Data Table - Spatial Averaging, 50 km cell.

4.12.3 Detailed Structure

Each 50 km cell can be identified by a pair of indices *l* and *m* that are proportional, respectively, to the y and x co-ordinates of the lower left-hand corner of the cell. Thus if these co-ordinates are *X* and *Y*, then

$$X = (m - 5)\,\Delta X; \quad (m = 0, 1, \ldots MAX\_CELLS\_X - 1)$$

$$Y = l\,\Delta Y; \quad (l = 0, 1, \ldots MAX\_CELLS\_Y - 1)$$

where ΔX and ΔY are the dimensions of the cell. All internal variables associated with the averaged product algorithms are duplicated for each cell, and should be imagined to be virtually present in memory at all times for the purpose of algorithm definition.

Each cell is further subdivided into 9 sub-cells of dimensions ($\Delta X/3$) and ($\Delta Y/3$). The sub-cells within each cell are identified by an index *k* in the range 0 to 8 as follows:

6	7	8
3	4	5
0	1	2

For the averaged channel values there are for each channel and for each view four cumulative sums, depending on surface type and cloud flag as follows:

sea, clear	sea, cloud
land, clear	land, cloud

We define a surface type flag = 0 (sea) or 1 (land) and a cloud state flag = 0 (clear) or 1 (cloud). For the purpose of indexing and identifying the AATSR channels, the following conventional numbering scheme will be adopted.

Channel	Symbol	Index
12 micron	ir12	1
11 micron	ir11	2
3.7 micron	ir37	3
1.6 micron	v16	4

0.870 micron	v870	5
0.670 micron	v670	6
0.55 micron	v555	7

Each average is defined by a sum of the form

$$A(ch,v;sf,cl,k,l,m) = \left\{\sum I(ch,v;i,j)\right\} \Big/ M(ch,v;sf,cl,k,l,m)$$

where the sum is over all valid pixels which fall within the cell, are filled (or not unfilled), are valid and have the correct cloud/surface type flag. That is, the sum is over all values of *i* and *j* such that all four of the following conditions are satisfied;

the pixel indexed by *i* and *j* falls within the sub-cell;

pixel_fill_state(*i*, *j*) is not unfilled;

I(ch, v; i, j) is valid

surface type, cloud state, flags at *i* and *j* have the correct value.

A total of 56 sums and averages are calculated. A separate group of totals are calculated for each channel and view (nadir and forward), there being 14 channel/view combinations. For each combination of channel and view four totals are maintained, and 4 averages computed, corresponding to the four combinations of the cloud state and surface type flags.

```
for l = 0, 1, ... while not end of data
for m = 0, MAX_CELLS_X - 1
```

Step 4.12.1 Derive channel totals for each cell

Perform steps 4.12. 1.1 to 4.12.1.7 for each pixel that falls within the cell identified by indices *l*, *m* unless the pixel is unfilled in both images:

```
for i' = 0, ΔY-1
for j' = 0, ΔX-1
```

$$i = \Delta Y \cdot l + i'$$

$$j = \Delta X \cdot m + j_0 + j'$$

(Req 4.12-1)

Step 4.12.1.1 Identify cell and subcell indices and pixel state

Identify the *i, j* co-ordinates of the lower left corner of the cell; these are the minimum values of *i, j* that are still greater than *lΔY, mΔX* + *MX* respectively.

cell_lat(l, m) = 1000000. * *image_latitude(i, j)*

cell_long(l, m) = 1000000. * *image_longitude(i, j)*

evaluated at those values of *i, j*.

The sub-cell index *k* is given by

k = 3 *[3 (*i′*/Δ*Y*)] + [3 (*j′*/Δ*X*)]

where the inclusion of a quantity in square brackets implies that the integer part is to be taken.

If the pixel identified by (*i, j*) is the first pixel (i.e. the leftmost) to fall within the cell *l, m*, ensure that all counters and cumulative sums are initialised to zero as follows:

S(ch, v; sf, cl, k, l, m) = 0.0

M(ch, v; sf, cl, k, l, m) = 0

N(v; sf, cl; k, l, m) = 0

band_sum(k, l, m) = 0

across-track_sum(sf; k, l, m) = 0 for each *k* = 0, 8.

If the pixel identified by (*i, j*) is the first pixel to fall within this sub-cell, assemble the cell geolocation and allied information as follows:

The time tag associated with the sub_cell [L2-INT-21] is derived from the scan number associated with the pixel *i, j*, and with the scan times from ADS #4. Starting from the indices *i, j*, derive the UTC time of the pixel in exactly the same way as described for the derivation of *utc*(*cell*) in Section 4.8.4 (Step 4.8.1.1), and assign it to *utc*(*k, l, m*) [L2-INT-21].

If k = 0 assign [L2-INT-20]

utc(*l, m*) = *utc*(*k, l, m*).

Assign the cell positional information:

sub_cell_lat(k, l, m) = 1000000. * *image_latitude(i, j)*

sub_cell_long(k, l, m) = 1000000. * *image_longitude(i, j)*

sub_cell_band(k, l, m) = band(j)

sub_cell_index(k, l, m) = i + [ΔY/6]

(This is used in the LST calculation, Section 4.14. Note that in practice [ΔY/6] equals 8, which is half the sub-cell dimension. The resulting along-track index refers to the mid-point of the sub-cell.)

nadir_solar_el(k, l, m) = nadir_band_centre_solar_elevation(i, band(j))

frwrd_solar_el(k, l, m) =frwrd_band_centre_solar_elevation(i, band(j))

mean_band(k, l, m) = band(j + ΔX/6)

If <view>*_band_centre_solar_elevation(i, band(j))* > 0.0 then

<view>*_day(k, l, m)* = TRUE

otherwise

<view>*_day(k, l, m)* = FALSE

where <view> = <*nadir* | *frwrd*>.

(Req 4.12-1)

Step 4.12.1.2 Process nadir Pixels

Perform steps 4.12.3.1.3 and 4.12.3.1.4 to process the nadir pixels unless the nadir pixel is unfilled (i.e. unless *nadir_fill_state(i, j)* = UNFILLED_PIXEL) or *m* (computed as above) is out of range.

Step 4.12.1.3 Identify the surface type and cloud state associated with the nadir pixel:

sf = 0 if nadir view land flag [L2-INT-232](*i, j*) = FALSE

sf = 1 if nadir view land flag [L2-INT-232](*i, j*) = TRUE

cl = 0 if nadir view cloud flag [L2-INT-233](*i, j*) = FALSE

cl = 1 if nadir view cloud flag [L2-INT-233](*i, j*) = TRUE.

Increment the pixel counters associated with the cloud state and surface type just determined:

$$N(n; sf, cl, k, l, m) \leftarrow N(n; sf, cl, k, l, m) + 1$$

If *cl* = 0 then increment the cumulative across-track index as follows:

$$across\_track\_sum(sf; k, l, m) \leftarrow across\_track\_sum(sf; k, l, m) + j$$

Step 4.12.1.4 Update nadir view channel totals

For each channel of the nadir view *ch* perform this step if the corresponding nadir pixel is valid:

$$S(ch, n; sf, cl, k, l, m) \leftarrow S(ch, n; sf, cl, k, l, m) + I(ch, n; i, j)$$

$$M(ch, n; sf, cl, k, l, m) \leftarrow M(ch, n; sf, cl, k, l, m) + 1$$

Step 4.12.1.5 Process forward Pixels:

Perform steps 4.12.3.1.6 and 4.12.3.1.7 to process the forward pixels unless the forward pixel is unfilled (i.e. unless *frwrd_fill_state*(*i, j*) = UNFILLED_PIXEL).

Step 4.12.1.6 Identify the surface type and cloud state associated with the forward pixel:

sf = 0 if frwrd cloud state land flag [L2-INT-248](*i, j*) = FALSE

sf = 1 if frwrd cloud state land flag [L2-INT-248](*i, j*) = TRUE

cl= 0 if frwrd cloud state cloud flag [L2-INT-249](*i, j*) = FALSE

cl = 1 if frwrd cloud state cloud flag [L2-INT-249](*i, j*) = TRUE

Increment the pixel counters associated with the cloud state and surface type just determined:

$$N(f; sf, cl, k, l, m) \leftarrow N(f; sf, cl, k, l, m) + 1$$

Step 4.12.1.7 Update forward view channel totals

For each channel of the forward view, perform the following steps if the corresponding forward pixel is valid:

$$S(ch, f; sf, cl, k, l, m) \leftarrow S(ch, f; sf, cl, k, l, m) + I(ch, f; i, j)$$

$$M(ch, f; sf, cl, k, l, m) \leftarrow M(ch, f; sf, cl, k, l, m) + 1$$

(Req 4.12-2)

```
end for (loop over j')
end for (loop over i')
```

Step 4.12.2 Derive average values.

When no more pixels remain to be added to the cell, or at the end of the data set, compute the averages. The following equation is evaluated for each channel (ch = ir12, ir11, ir37, v16, v870, v670, v555), for each view $v = n, f$, for surface type $sf = 0, 1$ and for cloud state $cl = 0, 1$

If $M\{ch, v; sf, cl, k, l, m\} > 0$

$$A(ch,v;sf,cl,k,l,m) = 10 \cdot S(ch,v;sf,cl,k,l,m)/M(ch,v;sf,cl,k,l,m)$$

(note the conversion to units of 0.001 K) otherwise set

$$A(ch,v;sf,cl,k,l,m) = -1.0$$

The mean in the larger (50 km) cell is given by

$$\tilde{A}(ch,v;sf,cl,l,m) = \frac{1}{\mu}\sum_k A(ch,v;sf,cl,k,l,m) \text{ if } \mu > 0$$

$$\tilde{A}(ch,v;sf,cl,l,m) = -1 \text{ if } \mu = 0$$

where the sum is over all $k \in \{0 \le k \le 8\}$ having a valid subcell mean A and μ is the number of such valid means. The number of pixels that contribute to the mean is similarly

$$\tilde{M}(ch,v;sf,cl,l,m) = \sum_k M(ch,v;sf,cl,k,l,m)$$

The standard deviation of the mean is

$$\sigma(ch,v;sf,cl,l,m) = \left\{\frac{1}{\mu-1}\sum_k (A(ch,v;sf,cl,k,l,m) - \tilde{A}(ch,v;sf,cl,l,m))^2\right\}^{1/2}$$

provided $\mu > 1$, otherwise set the standard deviation to –1. In all cases the sum is over sub-cells having valid means (i.e the number of contributing pixels M is positive).

(Req 4.12-3)

The mean across-track pixel number [L2-INT-459] is calculated for each sub-cell k = 0, 8 and for each surface type sf = 0, 1:

If *N(n; sf,* 0, *k, l, m)* > 0 then set

across_track_mean(sf, k, l, m) =

integer part of (*across_track_sum(sf; k, l, m)* / *N(n; sf,* 0, *k, l, m)*)

otherwise set

across_track_mean(sf, k, l, m) = -1

The mean to be associated with the 30 arc minute cell is [L2-INT-468] *across_track_mean(sf; l, m)* = $\frac{1}{\mu}\sum_{k} across\_track\_mean(sf;k,l.m)$ if μ > 0

across_track_mean(sf; l, m) = -1 if μ = 0

where the sum is over all $k \in \{0 \le k \le 8\}$ for which the number of contributing pixels

N(n; sf, 0, *k, l, m)* > 0

and μ is the number of valid *k*.

(Req 4.12-3.1)

Step 4.12.3 Derive Pixel Threshold Failure Flags Words (17 km cell)

For cell *l*, *m* and for each sub-cell *k* = 0, 9:

For surface type *sf* = 0, 1 and for the nadir view v = n:

[*PFF(n, sf; k, l, m)*](bit 0) = 1 if *M(ir12, n; sf, 0, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 1) = 1 if *M(ir11, n; sf, 0, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l. m)*](bit 2) = 1 if *M(ir37, n; sf, 0, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 3) = 1 if *M(v16, n; sf, 0, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 4) = 1 if *M(v870, n; sf, 0, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 5) = 1 if *M(v670, n; sf, 0, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 6) = 1 if *M(v555, n; sf, 0, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 7) = 1 if *M(ir12, n; sf, 1, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 8) = 1 if *M(ir11, n; sf, 1, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 9) = 1 if *M(ir37, n; sf, 1, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 10) = 1 if *M(v16, n; sf, 1, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 11) = 1 if *M(v870, n; sf, 1, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 12) = 1 if *M(v670, n; sf, 1, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 13) = 1 if *M(v555, n; sf, 1, k, l, m)* < [L2-AUX3-5], otherwise 0

[*PFF(n, sf; k, l, m)*](bit 14) = 1 if *nadir_day(k, l, m)*= TRUE, otherwise 0

[*PFF(n, sf; k, l, m)*](bit 15) = 0

(Req 4.12-4)

Similarly for surface type *sf* = 0, 1 and for the forward view v = f

Calculate the corresponding word *PFF(f, sf; k, l, m)*:

Set bits 0 to 13 inclusive as above, substituting the view index *f* in place of *n*, and substituting the forward threshold value [L2-AUX3-6] in place of [L2-AUX3-5].

[*PFF(f, sf; k, l, m)*](bit 14) = 1 if *frwrd_day(k, l, m)*= TRUE, otherwise 0

[*PFF(f, sf; k, l, m)*](bit 15) = 0

(Req 4.12-5)

Step 4.12.4 Derive Pixel Threshold Failure Flags Words (50 km cells)

For each cell *l, m*:

For surface type *sf* = 0, 1 and for the nadir view v = n:

[*PFF(n, sf; l, m)*](bit 0) = 1 if $\tilde{M}(ir12, n; sf, 0, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 1) = 1 if $\tilde{M}(ir11, n; sf, 0, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 2) = 1 if $\tilde{M}(ir37, n; sf, 0, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 3) = 1 if $\tilde{M}(v16, n; sf, 0, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 4) = 1 if $\tilde{M}(v870, n; sf, 0, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 5) = 1 if $\tilde{M}(v670, n; sf, 0, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 6) = 1 if $\tilde{M}(v555, n; sf, 0, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 7) = 1 if $\tilde{M}(ir12, n; sf, 1, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 8) = 1 if $\tilde{M}(ir11, n; sf, 1, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 9) = 1 if $\tilde{M}(ir37, n; sf, 1, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 10) = 1 if $\tilde{M}(v16, n; sf, 1, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 11) = 1 if $\tilde{M}(v870, n; sf, 1, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 12) = 1 if $\tilde{M}(v670, n; sf, 1, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 13) = 1 if $\tilde{M}(v555, n; sf, 1, l, m)$ < [L2-AUX3-7], otherwise 0

[*PFF(n, sf; l, m)*](bit 14) = 1 if *nadir_day(k, l, m)*= TRUE for some *k*, otherwise 0

[*PFF(n, sf; l, m)*](bit 15) = 0

(Req 4.12-6)

Similarly for surface type *sf* = 0, 1 and for the forward view v = f

Calculate the corresponding word *PFF(f, sf; l, m)*:

Set bits 0 to 13 inclusive as above, substituting the view index *f* in place of *n*, and substituting the forward threshold value [L2-AUX3-8] in place of [L2-AUX3-7].

[*PFF(f, sf; k, l, m)*](bit 14) = 1 if *frwrd_day(k, l, m)*= TRUE for some *k*, otherwise 0

[*PFF(f, sf; k, l, m)*](bit 15) = 0

Step 4.12.5. Derive pixel counts for cell

For each cell *l, m* and for each view *v* = *n, f*:

For each subcell *k* = 0, 8:

Total of filled pixels over land:

$$N\_land(v; k, l, m) = N(v; 1, 0, k, l, m) + N(v; 1, 1, k, l, m)$$

(Req 4.12-7)

Total of filled pixels over sea:

$$N\_sea(v; k, l, m) = N(v; 0, 0, k, l, m) + N(v; 0, 1, k, l, m)$$

(Req 4.12-8)

Total of filled pixels:

$$N\_total(v; k, l, m) = N\_land(v; k, l, m) + N\_sea(v; k, l, m)$$

(Req 4.12-9)

Derive cloudy pixel percentages for each sub-cell:

$$pcs(v; k, l, m) = 10000 * N(v; 0, 1, k, l, m)/N\_sea(v; k, l, m)$$

$$pcl(v; k, l, m) = 10000 * N(v; 1, 1, k, l, m)/N\_land(v; k, l, m)$$

(Req 4.12-10)

end for (*k*)

Derive aggregate counts:

Total of filled pixels over land:

$$N\_land(v; l, m) = \sum_{k=0}^{8} N\_land(v; k, l, m)$$

(Req 4.12-11)

Total of filled pixels over sea:

$$N\_sea(v; l, m) = \sum_{k=0}^{8} N\_sea(v; k, l, m)$$

(Req 4.12-12)

Total of filled pixels:

$$N\_total(v; l, m) = N\_land(v; l, m) + N\_sea(v; l, m)$$

(Req 4.12-13)

Derive cloudy pixel percentages for the cell:

$$pcs(v; l, m) = 10000 * (\sum_{k=0}^{8} N(v; 0, 1, k, l, m))/N\_sea(v; l, m)$$

$$pcl(v; l, m) = 10000 * (\sum_{k=0}^{8} N(v; 1, 1, k, l, m))/N\_land(v; l, m)$$

(Req 4.12-14)

end for (*l*, *m*, *v*)

4.13 Module Definition: Averaged SST Retrieval (50 km Cell)

4.13.1 Functional Description

This module derives the averaged SST from the averaged brightness temperatures determined using the module described in Section 4.12 above.

4.13.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-AUX2-1	a[i][j][0]	averaged sst retrieval a[3][38][0]	float	K/100	4	114
L2-AUX2-2	a[i][j][1]	averaged sst retrieval a[3][38][1]	float	none	4	114
L2-AUX2-3	a[i][j][2]	averaged sst retrieval a[3][38][2]	float	none	4	114
L2-AUX2-4	b[i][j][0]	averaged sst retrieval b[3][38][0]	float	K/100	4	114
L2-AUX2-5	b[i][j][1]	averaged sst retrieval b[3][38][1]	float	none	4	114
L2-AUX2-6	b[i][j][2]	averaged sst retrieval b[3][38][2]	float	none	4	114
L2-AUX2-7	b[i][j][3]	averaged sst retrieval b[3][38][3]	float	none	4	114
L2-AUX2-8	c[i][j][0]	averaged sst retrieval c[3][38][0]	float	K/100	4	114
L2-AUX2-9	c[i][j][1]	averaged sst retrieval c[3][38][1]	float	none	4	114
L2-AUX2-10	c[i][j][2]	averaged sst retrieval c[3][38][2]	float	none	4	114
L2-AUX2-11	c[i][j][3]	averaged sst retrieval c[3][38][3]	float	none	4	114
L2-AUX2-12	c[i][j][4]	averaged sst retrieval c[3][38][4]	float	none	4	114
L2-AUX2-13	d[i][j][0]	averaged sst retrieval d[3][38][0]	float	K/100	4	114
L2-AUX2-14	d[i][j][1]	averaged sst retrieval d[3][38][1]	float	none	4	114
L2-AUX2-15	d[i][j][2]	averaged sst retrieval d[3][38][2]	float	none	4	114
L2-AUX2-16	d[i][j][3]	averaged sst retrieval d[3][38][3]	float	none	4	114
L2-AUX2-17	d[i][j][4]	averaged sst retrieval d[3][38][4]	float	none	4	114
L2-AUX2-18	d[i][j][5]	averaged sst retrieval d[3][38][5]	float	none	4	114
L2-AUX2-19	d[i][j][6]	averaged sst retrieval d[3][38][6]	float	none	4	114
L2-AUX6-1	j	pixel index (j)	ss	none	2	512
L2-AUX6-2	map(j)	Across-track band index (map index)	ss	none	2	512
L2-AUX3-10		AST Cell dime nsion	sl	none	4	1
L2-AUX3-11		TROPICAL_INDEX	float	degrees	4	1
L2-AUX3-12		TEMPERATE_INDEX	float	degrees	4	1
L2-AUX3-13		POLAR_INDEX	float	degrees	4	1
L2-AUX3-14		NADIR-PIXELS_THRESH	sl	none	4	1
L2-AUX3-15		FRWRD_PIXELS_THRESH	sl	none	4	1
L2-AUX3-16		IR37_THRESH	float	none	4	1

Table 4.13.1: Input Data Table - Averaged SST Retrieval

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-INT-28	utc(l, m)	UTC for cell l, m	double	days	8	per cell
L2-INT-61	utc(k, l, m)	sub-cell UTC	double	days	8	k = 0, 8
L2-INT-62	sub_cell_lat(k, l, m)	sub-cell latitude	ss	cell number		
L2-INT-63	sub_cell_long(k, l, m)	sub-cell longitude	ss	cell number		
L2-INT-457	mean_band(k, l, m)	mean across-track band number	ss	none	2	k = 0, 8
L2-INT-66	S(ch, v; sf, cl, k, l, m)	sub-cell total, ch = 1, ..., 7	sl	n/a	4	
L2-INT-67	M(ch, v; sf, cl, k, l, m)	sub-cell pixel count, ch = 1, ..., 7	ss	none	2	
L2-INT-68	A(ch, v; sf, cl, k, l, m)	sub-cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	ss	0.01K	2	
L2-INT-70	$\tilde{M}$ (ch, v; sf, cl, l, m)	cell pixel count, ch = 1, ..., 7	ss	none	2	
L2-INT-71	$\tilde{A}$ (ch, v; sf, cl, l, m)	cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	ss	0.01K	2	

L2-INT-73	σ(ch, v; sf, cl, l, m)	standard deviation of the cell average	float	0.01K or 0.01%	4	
L2-INT-79	nadir_day(k, l, m)	nadir view sub-cell day/night flag	ss	flag	2	k = 0, 8
L2-INT-80	frwrd_day(k, l, m)	forward view sub-cell day/night flag	ss	flag	2	k = 0, 8
L2-INT-75	nadir_solar_el(k, l, m)	nadir solar elevation for sub-cell	float	degrees	4	
L2-INT-76	frwrd_solar_el(k, l, m)	frwrd solar elevation for sub-cell	float	degrees	4	
local	i	index to latitude zone; i = 0, 1, 2	sl	none	4	
local	j	index to across-track bands j = 0, 9	sl	none	4	
local	q	index to coefficient set	sl	none	4	
L2-INT-459	across_track_mean(sf; k, l, m)	mean across-track pixel index, subcell k	ss	none	2	k = 0, 8
L2-INT-460	a(i, j, q)	averaged sst retrieval a coefficients	float	mixed	4	342
L2-INT-461	b(i, j, q)	averaged sst retrieval b coefficients	float	mixed	4	456
L2-INT-462	c(i, j, q)	averaged sst retrieval c coefficients	float	mixed	4	570
L2-INT-463	d(i, j, q)	averaged sst retrieval d coefficients	float	mixed	4	798
L2-INT-466		nadir_asst_uses_ir37(k, l, m)	ss	flag	2	k = 0, 8
L2-INT-467		dual_asst_uses_ir37(k, l, m)	ss	flag	2	k = 0, 8
L2-INT-469	sst_mean_pixel(sf; l, m)	mean across-track pixel index, cell l, m	ss	none	2	per cell
L2-INT-61	map(j)	across-track mapping	sl	none	4	j = 0, 512
	minpn	minimum nadir pixels	float	none	4	
	minpf	minimum forward pixels	float	none	4	
local	latitude	temporary latitude	float	degrees	4	
L2-INT-83	T_nadir(l, m)	nadir view sst	ss	0.01K	2	
L2-INT-84	T_nadir(k, l, m)	sub_cell nadir view sst[9]	ss	0.01K	2	
L2-INT-85	T_dual(l, m)	dual view sst	ss	0.01K	2	
L2-INT-86	T_dual(k, l, m)	sub_cell dual view sst[9]	ss	0.01K	2	
L2-INT-87	σ_nadir(ASST; l, m)	standard deviation of nadir view ASST	ss	0.01K	2	
L2-INT-88	σ_dual(ASST; l, m)	standard deviation of dual view ASST	ss	0.01K	2	
L2-INT-464	ast_conf(0; k, l, m)	AST confidence word for sea sub-cell	sl	flags	4	k = 0, 8
L2-INT-465	ast_conf(0; l, m)	AST confidence word for sea cell l, m	sl	flags	4	per cell
local	T0	tropical_sst	float	deg.	4	
local	T1	temperate_sst	float	deg.	4	
local	T2	polar_sst	float	deg.	4	
local	w	interpolation weight	float	none	4	
local	μ_1	number of sub-cells in cell average (nadir)	sl	none	4	
local	μ_2	number of sub-cells in cell average (dual)	sl	none	4	

Table 4.13-2: Internal Data Table - Averaged SST Retrieval

4.13.3 Detailed Structure

Both dual-view and nadir only sea surface temperatures are derived.

In the processing, each half-degree cell is represented by a structure containing, or is associated with, the necessary intermediate and output variables including the averaged brightness temperatures for each 10 arc-minute cell contained within the larger cell. All cells should be virtually present in memory, but how this is achieved is a matter for the implementer.

Note on notation: In the following we adopt the following abbreviated notation for the average brightness temperatures.

$$T_{ch}^{nadir} = \left[\frac{\text{float}(S(ch,n;\,0,0,k,l,m))}{\text{float}(M(ch,n;\,0,0,k,l,m))} \right]$$

$$T_{ch}^{frwrd} = \left[\frac{\text{float}(S(ch, f; 0,0,k,l,m))}{\text{float}(M(ch, f; 0,0,k,l,m))} \right]$$

where *ch* indicates one of the seven channels. This notation is adopted to reduce the proliferation of indices; note that where it is used, a dependence on *k*, *l* and *m* is implied. Note also that the above quantities must be computed using a floating point computation, although *S* and *M* are of type integer, to ensure that sufficient precision is maintained. Substitution of the quantities *A*(*ch, f; 0, 0, k, l, m*) would not ensure this.

Processing is applied to cells for which the processing of Step 4.12.1 is complete; i.e. no more pixels remain to be added to the cell.

The processing to derive averaged SST is done as follows:

Step 4.13.1 Read look-up tables.

On first entry, input the look-up tables of averaged SST retrieval coefficients.

This is done once at initialisation. Retrieval coefficients are specified for three latitude zones (tropical, temperate and polar) and for 38 bands or strips running parallel to the ground track, and corresponding to different viewing angles. Distinct sets of coefficients are supplied for day/night and for nadir only/dual view retrievals, as follows.

Index	Zone	Set	Application
0	tropical	a	nadir only, day
1	temperate	b	nadir only, night
2	polar	c	dual view, day
		d	dual view, night

Before reading the retrieval coefficients ensure that the mapping array *map*(*j*) [L2-INT-61] is available. This has been read in during Step 4.6.1. If it is desired to re-input it independently in this module then proceed as before:. Open the data set L2-AUX6 and set

map(j) = [L2-AUX6-2](j), j = 0, 511

(Note that [L2-AUX6-1](j) = j, j = 0, 511.)

Open the file of retrieval coefficients to access data set L2-AUX2.

For each latitude zone *i* = 0, 1, 2 (outer loop) and for each across-track band *j* = 0 to 37 (inner loop);

Read in next record of file.
Extract the *a* coefficients [L2-INT-460] as follows
a(i, j, 0) = [L2-AUX2-1]
a(i, j, 1) = [L2-AUX2-2]
a(i, j, 2) = [L2-AUX2-3]

Similarly extract in the *b*, *c* and *d* sets of coefficients:
b(i, j, q) = [L2-AUX2-<4 + q>], *q* = 0, 1, 2, 3;
c(i, j, q) = [L2-AUX2-<8 + q>], *q* = 0, 1, 2, 3 4;
d(i, j, q) = [L2-AUX2-<13 + q>], *q* = 0, 1, 2, 3, 4, 5, 6.

(Note that if the retrieval coefficients have already been read in at Step 4.9.1 and are still available, they need not be read again. Instead the values [L2-INT-460] to [L2-INT-363] may be equated to [L2-INT-360] to [L2-INT-363] respectively. There is only one set of averaged retrieval coefficients.)

(Req 4.13-1)

Step 4.13.2 Nadir view average.

Calculate the nadir view averaged SST value for each of the 10-arcmin cells. Note that in the following, if the flags *nadir_asst_uses_ir37, dual_asst_uses_ir37 are initialised to the value FALSE, then Reqs. 4-13-2a, 4-13-9a are logivally redundant.*

Step 4.13.2.1

Determine the minimum number of pixels required for the cell, for the nadir. This is

minpn = 340 * *NADIR_PIXELS_THRESH* + 1.

(No latitude dependence is required here as sub-cell size is independent of latitude.)

If *M(ir12, n*; 0, 0, *k, l, m)* ≥ *minpn* and *M(ir11, n*; 0, 0, *k, l. m)* ≥ *minpn* proceed to calculate the retrieved sst as below, otherwise set

T_nadir(k, l, m) = -1.0

(Req 4.13-2)

nadir_asst_uses_ir37(*k, l, m*) = *FALSE*

(Req 4-13-2a)

Step 4.13.2.2

For night-time data, if *nadir_day(k, l, m)* = FALSE, test whether the ratio of pixels with valid 3.7 µm data is greater or less than the threshold value and use the appropriate (two or three channel) SST algorithm. The 3.7 micron channel is valid if

float{*M(ir37, n; 0, 0, k, l, m)*} / float{*M(ir11, n; 0, 0, k, l, m)*} ≥ *IR37_THRESH.*

Otherwise use the two-channel algorithm. The two-channel algorithm is always used for day-time data, that is, if *nadir_day(k, l, m)* = TRUE.

Step 4.13.2.3

Calculate the averaged SST using the nadir-view retrieval coefficients for the appropriate across-track band given by *j* = *map*(*across_track_mean(0; k, l, m)* and for the two or three channel algorithm as appropriate, for each latitude zone *i* = 0, 1, 2:

Step 4.13.2.3.1

Perform this step if the 3.7 micron channel is not available for use.

The equation for use with the nadir view is

$$T_{sst,i}^{nadir} = 100.0a_0 + a_1 T_{ir11}^{nadir} + a_2 T_{ir12}^{nadir}$$

where

$$a_q = a(i, j, q).$$

(Req 4.13-3)

(Here and elsewhere in this module the factor of 100 is to ensure consistency of units between the brightness temperatures, in units of 0.01K, and the leading coefficient, in K.)

Set

nadir_asst_uses_ir37(k, l, m) = FALSE

(Req 4.13-3.1)

Step 4.13.2.3.2

Perform this step if the 3.7 micron channel is to be used.

The equation for use with the nadir view is

$$T_{sst,i}^{nadir} = 100.0 b_0 + b_1 T_{ir11}^{nadir} + b_2 T_{ir12}^{nadir} + b_3 T_{ir37}^{nadir}$$

where

$$b_q = b(i, j, q).$$

(Req 4.13-4)

(As before, the factor of 100 is to ensure consistency of units.)

Set

nadir_asst_uses_ir37(k, l, m) = TRUE

(Req 4.13-4.1)

Step 4.13.2.4

Return latitude-corrected SST (with linear interpolation).

If the cell is in the polar or tropical zone, return the corresponding retrieval. If the cell is in the temperate zone, use linear interpolation with respect to latitude to derive the averaged SST from the values for the temperate zone and the appropriate adjacent zone.

If abs(*latitude*) < TROPICAL_INDEX then

$$T\_nadir(k,l,m) = T_{sst,0}^{nadir}$$ (Req 4.13-5)

If TROPICAL_INDEX ≤ abs(*latitude*) < TEMPERATE_INDEX, the final value for the retrieved sst is given by

$$T\_nadir(k,l,m) = T_{sst,0}^{nadir} + w \cdot (T_{sst,1}^{nadir} - T_{sst,0}^{nadir})$$ (Req 4.13-6)

where

$$w = \frac{(\text{abs}(latitude) - TROPICAL\_INDEX)}{(TEMPERATE\_INDEX - TROPICAL\_INDEX)}$$

If the TEMPERATE_INDEX ≤ abs(latitude) < POLAR_INDEX the but not less than, the final value for the retrieved sst is given by

$$T\_nadir(k,l,m) = T_{sst,1}^{nadir} + w \cdot (T_{sst,2}^{nadir} - T_{sst,1}^{nadir})$$ (Req 4.13-7)

where

$$w = \frac{(\text{abs}(latitude) - TEMPERATE\_INDEX)}{(POLAR\_INDEX - TEMPERATE\_INDEX)}$$

If POLAR_INDEX ≤ abs(*latitude*)

$$T\_nadir(k,l,m) = T_{sst,2}^{nadir}$$ (Req 4.13-8)

Step 4.13.3 Dual view average.

Calculate the dual view averaged SST value for the 10-arcmin cells.

Step 4.13.3.1

Determine the minimum numbers of pixels required for the cell, for both. The threshold for the nadir view is *minpn* calculated as above. That for the forward view is

minpf = 340 * *FRWRD_PIXELS_THRESH* + 1.

If the number of valid pixels in the either view is less than the threshold value calculated, move to the next 17 km cell.

If

M(ir12, n; 0, 0, *k, l,m)* ≥ *minpn* and *M(ir11, n*;, 0, 0, *k, l, m)* ≥ *minpn*

and

M(ir12, f, 0, 0, *k, l, m)* ≥ *minpf* and *M(ir11, f*, 0, 0, *k, l, m)* ≥ *minpf*

proceed to calculate the retrieved sst as below, otherwise set

T_dual(k, l, m) = -1.

(Req 4.13-9)

dual_asst_uses_ir37(*k, l, m*) = *FALSE*

(Req 4-13-9a)

Step 4.13.3.2

For night-time data, defined by the condition
(*nadir_day(k, l, m)* = FALSE and *frwrd_day(k, l, m)* = FALSE),
test whether the ratio of pixels with valid 3.7 µm data in the two views is greater or less than the threshold value and use the appropriate (two or three channel) SST algorithm. The 3.7 micron channel is valid if

float{*M(ir37, n; 0, 0, k, l, m)* + *M(ir37, f; 0, 0, k, l, m)*} /
float{*M(ir11, n; 0, 0, k, l, m)* + *M(ir11, f; 0, 0, k, l, m)*} ≥ *IR37_THRESH*.

Otherwise use the two-channel algorithm. The two-channel algorithm is always used for day-time data, defined by the condition
(*nadir_day(k, l, m)* = TRUE or *frwrd_day(k, l, m)* = TRUE).

(Req 4.13-10)

Step 4.13.3.3

Calculate the averaged SST using the dual-view retrieval coefficients for the appropriate across-track band given by *j* = *map*(*across_track_mean(0; k, l, m)* and for the two or three channel algorithm as appropriate, for each latitude zone.

(Req 4.13-11)

Step 4.13.3.3.1

Perform this step if the 3.7 micron channel is not available for use.

The algorithm using both views is given by

$$T_{sst,i}^{dual} = 100.0c_0 + c_1 T_{ir11}^{nadir} + c_2 T_{ir12}^{nadir} + c_3 T_{ir11}^{frwrd} + c_4 T_{ir12}^{frwrd}$$ (Req 4.13-12)

where

$$c_q = c(i, j, q).$$

Set

dual_asst_uses_ir37(k, l, m) = FALSE

(Req 4.13-12.1)

Step 4.13.3.3.2

Perform this step if the 3.7 micron channel is to be used.

$$T_{sst,i}^{dual} = 100.0d_0 + d_1 T_{ir11}^{nadir} + d_2 T_{ir12}^{nadir} + d_3 T_{ir37}^{nadir} + d_4 T_{ir11}^{frwrd} + d_5 T_{ir12}^{frwrd} + d_6 T_{ir37}^{frwrd}$$

(Req 4.13-13)

where

$$d_q = d(i, j, q).$$

Set

dual_asst_uses_ir37(k, l, m) = TRUE

(Req 4.13-13.1)

Step 4.13.3.4

Return latitude-corrected SST (with linear interpolation).

If abs(*latitude*) < TROPICAL_INDEX then

$$T\_dual(k,l,m) = T_{sst,0}^{dual}$$ (Req 4.13-14)

If TROPICAL_INDEX ≤ abs(*latitude*) < TEMPERATE_INDEX, the final value for the retrieved sst is given by

$$T\_dual(k,l,m) = T_{sst,0}^{dual} + w \cdot (T_{sst,1}^{dual} - T_{sst,0}^{dual})$$ (Req 4.13-15)

where

$$w = \frac{(\mathrm{abs}(latitude) - TROPICAL\_INDEX)}{(TEMPERATE\_INDEX - TROPICAL\_INDEX)}$$

If the TEMPERATE_INDEX ≤ abs(latitude) < POLAR_INDEX the but not less than, the final value for the retrieved sst is given by

$$T\_dual(k,l,m) = T_{sst,1}^{dual} + w \cdot (T_{sst,2}^{dual} - T_{sst,1}^{dual}) \qquad \text{(Req 4.13-16)}$$

where

$$w = \frac{(\text{abs}(latitude) - TEMPERATE\_INDEX)}{(POLAR\_INDEX - TEMPERATE\_INDEX)}$$

If POLAR_INDEX ≤ abs(*latitude*)

$$T\_dual(k,l,m) = T_{sst,2}^{dual} \qquad \text{(Req 4.13-17)}$$

Step 4.13.4

From up to nine 17 km cells within the 50 km cell, derive the mean nadir view SST for the 50 km cell, and the standard deviation of the 17 km SST values. Repeat for the dual-view retrieval. That is

$$T\_nadir(l,m) = \frac{1}{\mu_1}\sum_k T\_nadir(k,l,m)$$

$$T\_dual(l,m) = \frac{1}{\mu_2}\sum_k T\_dual(k,l,m) \qquad \text{(Req 4.13-18)}$$

where in each case the sum is over all values of k for which the respective sub-cell temperature is valid (i.e. has a positive value), and where μ1 and μ2 are the numbers of such valid temperatures in the nadir and forward views respectively. If either of the values μ_1 or μ_2 is zero, set the corresponding temperature to –1.

The mean across-track pixel number to be associated with the 50 km SST is [L2-INT-469]

$$sst\_mean\_pixel(0; l, m) = \frac{1}{\mu_1}\sum_k across\_track\_mean(0;k,l,m) \text{ if } \mu_1 > 0$$

$$sst\_mean\_pixel(0; l, m) = -1 \text{ if } \mu_1 = 0$$

where the sum is over all $k \in \{0 \le k \le 8\}$ for which corresponding SST *T_nadir*(*k*, *l*, *m*) is valid (not equal to –1.)

Step 4.13.5 Prepare the confidence flag word for the cell.

The confidence flag word for the sub-cell indexed by (*k*, *l*, *m*) should be prepared as follows:

Set bit 0 if 3-channel algorithm was used at Step 4.13.2.2.2, i.e. if
nadir_asst_uses_ir37(k, l, m) = *TRUE*, otherwise clear bit.

Set bit 1 if 3-channel algorithm was used at Step 4.13.2.3.2, i.e. if
dual_asst_uses_ir37(k, l, m) = *TRUE*, otherwise clear bit.

Set bit 2 if *nadir_day(k, l, m)* from §4.12.3 is TRUE, otherwise clear bit.

Set bit 3 if *frwrd_day(k, l, m)* from §4.12.3 is TRUE, otherwise clear bit.

The confidence flag word for the half-degree cell indexed by *l*, *m* will be derived by ORing together the words for those sub-cells (*k*, *l*, *m*), *k* = 0, ...8, for which a valid temperature was derived.

(Req 4.13-19)

4.14 Module Definition: Averaged LST and NDVI Retrieval (50 km Cell)

4.14.1 Functional Description

The Land Surface Temperature (LST) and Normalised Difference Vegetation Index (NDVI) are calculated for each sub-cell for which average reflectances over land have been calculated. The averaged LST and NDVI over all the subcells, and their standard deviation, are also computed.

4.14.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-AUX3-14		NADIR_PIXELS_THRESH	float	none	4	1
L2-AUX5-1		Coefficient A0 (day-time) for LST	float	K	4	1
L2-AUX5-2		Coefficient A1 (day-time) for LST	float	none	4	1
L2-AUX5-3		Coefficient A2 (day-time) for LST	float	none	4	1
L2-AUX5-4		Coefficient A0 (night-time) for LST	float	K	4	1
L2-AUX5-5		Coefficient A1 (night-time) for LST	float	none	4	1
L2-AUX5-6		Coefficient A2 (night-time) for LST	float	none	4	1
L2-AUX6-1		Vegetation class index [360][720] for LST	ss	n/a	2	720
L2-AUX7-1		Vegetation fraction[12][360][720]	ss	0.001	2	720
L2-AUX8-1		Precipitable water[12][360][720]	ss	0.01 mm	2	720
L2-AUX9-1		Topographic Variance Flag[360][720]	ss	n/a	2	720
L2-AUX10-1	d	Water vapour factor for LST retrieval	float	none	4	1
L2-AUX10-2	m	Angle factor for LST retrieval	ss	none	2	1
L2-AUX10-3	N_CLASS	Number of vegetation classes for LST	ss	none	2	1

Table 4.14-1: Input Data Table - LST Retrieval LUTs and auxiliary parameters

Parameter ID	Variable	Name	Type	Units	Field size	Fields
	k	sub-cell number k = 0, ..., 8	sl	none	4	1
	l	along-track 50 km cell index	sl	none	4	1
	m	across-track 50 km cell index	sl	none	4	1
		sub-cell total ch = 1, ..., 7	sl	n/a	4	
L2-INT-69	A(v870, n; 1, 0, k, l,m)	sub-cell reflectance average, 0.87 micron channel, nadir view, clear, land	ss	0.01%	2	
L2-INT-67	M(v870, n; 1, 0, k, l, m)	sub-cell pixel count, 0.87 micron channel, nadir view, clear, land	ss	none	2	
L2-INT-69	A(v670, n; 1, 0, k, l, m)	sub-cell reflectance average, 0.670 micron channel, nadir view, clear, land	ss	0.01%	2	
L2-INT-67	M(v670, n; 1, 0, k, l, m)	sub-cell pixel count, 0.670 micron channel, nadir view, clear, land	ss	none	2	
L2-INT-95	NDVI(k, l, m)	Averaged NDVI in 10-arcmin cells	ss	0.0001	2	k = 0, 8
L2-INT-96	<NDVI>(l, m)	mean NDVI	ss	0.0001	2	per cell
L2-INT-97	σ(NDVI; l, m)	standard deviation of NDVI	ss	0.0001	2	per cell
local	μ	number of sub-cells contributing to cell mean	sl	none	4	
L2-INT-98	N0(l, m)	Number of pixels in NDVI average, half degree cell	us	none	2	per cell

L2-INT-99	N1(k, l, m)	Number of pixels in NDVI average, 10 arc min cells	us	none	2	k = 0, 8
L2-INT-464	ast_conf(1; k, l, m)	AST confidence word for land sub-cell	sl	flags	4	k = 0, 8
L2-INT-465	ast_conf(1; l, m)	AST confidence word for land cell l, m	sl	flags	4	per cell
	The following parameters are required by the Land Surface Temperature algorithm:					
L2-INT-26	time(sg)	scan UTC	double	days	8	per sg
L2-INT-77	cell_lat(l, m)	cell latitude (50 km)	sl	μdeg	4	per cell
L2-INT-78	cell_long(l, m)	cell longitude (50 km)	sl	μdeg	4	per cell
L2-INT-470		vegetation_class(lat_index, lon_index)	ss	none	2	360 × 720
L2-INT-471		vegetation_fraction(lat_index, lon_index)	ss	0.001	4	360 × 720
L2-INT-472		precipitable_water(lat_index, lon_index)	ss	0.01 mm	2	360 × 720
L2-INT-473		Topographic_flag(lat_index, lon_index)	ss	none	2	360 × 720
L2-INT-474	lat_index	Latitude index: lat_index = 0, …359	sl	none	4	1
L2-INT-475	lon_index	Longitude index: lon_index = 0, 719	sl	none	4	1
L2-INT-489	disp_lat_index	Displaced latitude index: = 0, …359	sl	none	4	1
L2-INT-490	disp_lon_index	Displaced longitude index: = 0, 719	sl	none	4	1
L2-INT-476	month	Index to month: month = 0, 11	sl	none	4	1
L2-INT-477	sun_elev	Solar elevation at land pixel	float	degrees	4	1
L2-INT-478	sat_elev	Satellite elevation at land pixel	float	degrees	4	1
L2-INT-479	night	Day/night flag	sl	none	2	1
L2-INT-480	n	Non-linear exponent	float	none	4	1
L2-INT-481	f	Vegetation fraction at pixel	float	none	4	1
L2-INT-482	pw	Precipitable water at pixel	float	cm	4	1
L2-INT-482	coeff(class, i, j, 0)	Sub-array of coefficients A0	float	0.01K	4	64
L2-INT-484	coeff(class, i, j, 1)	Sub-array of coefficients A1	float	none	4	64
L2-INT-485	coeff(class, i, j. 2)	Sub-array of coefficients A2	float	none	4	64
L2-INT-486	w	Interpolation weight	float	none	4	1
L2-INT-487	a(k)	Retrieval coefficients for pixel	float	mixed	4	1
L2-INT-488	lst	Land surface temperature at pixel	float	0.01K	4	1
Local	lst	Land surface temperature at pixel	float	0.01K	4	1
Local	pw00	Precipitable water sample value	ss	0.01 mm	2	1
Local	pw01	Precipitable water sample value	ss	0.01 mm	2	1
Local	pw10	Precipitable water sample value	ss	0.01 mm	2	1
Local	pw11	Precipitable water sample value	ss	0.01 mm	2	1
Local	q	Latitude argument for bilinear interpolation	float	none	4	1
Local	p	Longitude argument for bilinear interpolation	float	none	4	1
Local	class	Index to table of cefficients	sl	none	4	1
Local	latitude	Temporary latitude value	float	Degrees	4	1
Local	minpn	Minimum nadir pixels	float	none	4	1
Local	kx	Sub-cell index in longitude	sl	none	4	1
Local	ky	Sub-cell index in latitude	sl	none	4	1
L2-INT-62	sub_cell_lat(k, l, m)	sub-cell latitude (17 km)	sl	μdeg	4	k = 0, 8
L2-INT-79	nadir_day(k, l, m)	nadir view day/night flag (17 km)	ss	flag	2	k = 0, 8
L2-INT-80	frwrd_day(k, l, m)	forward view day/night flag (17 km)	ss	flag	2	k = 0, 8
L2-INT-60	band(j)	number of across track band (or strip)	sl	none	4	j = 0, 511
L2-INT-121		nadir_band_edge_satellite_elevation(i, k)	float	degrees	4	k = 0, 10
L2-INT-459	across_track_mean(sf; k, l, m)	mean across-track pixel index, subcell k	ss	none	2	k = 0, 8
L2-INT-495	sub_cell_index(k, l, m)	Along-track index representative of sub-cell	sl	none	4	k = 0, 8
L2-INT-496	T_land(k, l, m)	Land surface temperature in sub-cell k.	ss	0.01 K	4	k = 0, 8
L2-INT-497	T_land(l, m)	Averaged land surface temperature in cell.	ss	0.01 K	4	per cell
L2-INT-498	σ_land(l, m)	Standard deviation of Averaged LST	ss	0.01 K	4	per cell
L2-INT-464	ast_conf(sf; k, l, m)	AST confidence word for sub-cell	sl	flags	4	k = 0, 8
L2-INT-465	ast_conf(sf; l, m)	AST confidence word for cell	sl	flags	4	per cell
L2-INT-469	sst_mean_pixel(sf; l, m)	Mean across-track pixel index, cell	ss	none	2	per cell

Table 4.14-2: Internal Data Table - Averaged NDVI Retrieval

4.14.3 Detailed Structure

The following processing is applied to cells after the processing of Step 4.12.1 is complete; i.e. no more pixels remain to be added to the cell.

Step 4.14.1 Calculate subcell NDVIs.

NDVI is defined by

$$NDVI(k,l,m) = 10000\,\frac{A(v870,n;1,0,k,l,m) - A(v670,n;1,0,k,l,m)}{A(v870,n;1,0,k,l,m) + A(v670,n;1,0,k,l,m)}$$

(Req 4.14-1)

provided both values are valid (not exceptional). Otherwise

NDVI(*k, l, m*) = -19999.

The number of pixels contributing to the sub-cell mean, *N*1(*k, l, m*), provided as a confidence indicator, is the smaller of *M(v870, n; 1, 0, k, l, m)* and *M(v670, n; 1, 0, k, l, m)*.

Step 4.14.2 Calculate cell NDVI.

The mean in the larger (50 km) cell is given by

$$\langle NDVI \rangle(l,m) = \frac{1}{\mu}\sum_k NDVI(k,l,m)$$ (Req 4.14-2)

where the sum is over all $k \in \{0 \le k \le 8\}$ having a valid subcell mean *NDVI* and μ is the number of such valid means. The number of pixels that contribute to the mean is similarly the smaller of

$$\tilde{M}(v870,n;1,0,l,m),\ \tilde{M}(v670,n;1,0,l,m).$$

The standard deviation of the mean is

$$\sigma(NDVI;l,m) = \left\{\frac{1}{\mu - 1}\sum_k (NDVI(k,l,m) - \langle NDVI \rangle(l,m))^2\right\}^{1/2}$$ (Req 4.14-3)

in all cases the sum is over sub-cells having valid means.

If the number of valid subcell means μ is zero, set

<*NDVI*>(*l, m*) = -19999.

(Req 4.14-4)

If the number of valid subcell means μ ≤ 1, so that a valid standard deviation cannot be calculated, set set

$$\sigma(NDVI;l,m) = -19999.$$

(Req 4.14-4.1)

Step 4.14.3 Read in coefficients and auxiliary tables for LST retrieval.

The coefficients for LST retrieval are identical to those used for the full resolution product, as read in in Step 4.6.1.2 (Section 4.6.3). If these coefficients are still available in the processor, there is no need to repeat the following steps.

Step 4.14.3.1 Read in coefficients

For each of the *N_CLASS* vegetation classes there are two records, for vegetation and for bare soil. Open the file of retrieval coefficients L2-AUX5.

The LST coefficient set is read in as follows.

for *class* = 0, *N_CLASS* - 1 (outer loop)
    for *i* = 0, 1 (inner loop)
        coeff(class, i, 0, 0) = [L2-AUX5-1]
        coeff(class, i, 0, 1) = [L2-AUX5-2]
        coeff(class, i, 0, 2) = [L2-AUX5-3]
        coeff(class, i, 1, 0) = [L2-AUX5-4]
        coeff(class, i, 1, 1) = [L2-AUX5-5]
        coeff(class, i, 1, 2) = [L2-AUX5-6]

(Req 4.14-5)

Step 4.14.3.2 Determine month index

Using a suitable calendar function, determine the month (*month* = 0, … 11) in which the data was collected from the scan time of start of data *time*(0)=[L2-INT-26](0):

month = *month*(*time*(0))

(Req 4.14-6)

Step 4.14.3.3 Read in auxiliary files

Note that in the cases of data sets L2-AUX7 and L2-AUX8 only one plane of data, that corresponding to the current month, is required in memory for a given run of the processor.

Read in Vegetation Class Index: Open the vegetation class file L2-AUX6.

for each latitude index *i* = 0, 359
        vegetation_class(*i*, *j*) = [L2-AUX6-1](*j*) for all *j* of record *i*.

(Req 4.14-7)

Read in Vegetation Fraction Table: Open the file of vegetation fraction data L2-AUX7.

for each latitude index *i* = 0, 359
        select record (360 × *month* + *i*)
        vegetation_fraction(*i*, *j*) = [L2-AUX7-1](*j*) for all *j* of selected record.

(Req 4.14-8)

Read in Precipitable Water Data: Open the file of precipitable water data L2-AUX8.

for each latitude index *i* = 0, 359
        select record (360 × *month* + *i*)
        precipitable_water(*i*, *j*) = [L2-AUX8-1](*j*) for all *j* of selected record.

(Req 4.14-9)

Read in Topographic Variance Flag: Open the file of topographic variance flags L2-AUX9.

for each latitude index *i* = 0, 359
        topographic_flag(*i*, *j*) = [L2-AUX9-1](*j*) for all *j* of record *i*.

(Req 4.14-10)

Step 4.14.4 Derive Land Surface Temperature for sub-cells

LST retrievals use the nadir view 11 and 12 micron channels in conjunction with retrieval coefficients derived from the tables.

Note that as in Section 4.13 we adopt an abbreviated notation for the average brightness temperatures in this section.

$$T_{ch,sf}^{nadir} = \left[\frac{\text{float}(S(ch,n;sf,0,k,cell))}{\text{float}(M(ch,n;sf,0,k,cell))} \right]$$

where *ch* indicates one of the long-wavelength infra-red channels, and where *sf* is the surface type flag. This notation is slightly more complex than that used in Section 4.13 because it is necessary to distinguish between land and sea averages. Where this notation it is used, a dependence on *k* and *l, m* is implied. As in section 4.13, these quantities must be computed using a floating point computation, although *S* and *M* are of type integer, to ensure that sufficient precision is maintained.

The calculation proceeds as follows for each cell in turn. Steps 4.14.4.1 to 4.14.4.5 are repeated for each sub-cell $k = 0, \ldots 8$ of the cell.

Step 4.14.4.1 Determine latitude and longitude indices

For the current sub-cell *k*, calculate

lat_index = integer part of [*cell_lat*(*l, m*)/500000.0] + 180

lon_index = integer part of [*cell_lon*(*l, m*)/500000.0] + 360

`(Req 4.14-11)`

disp_lat_index = integer part of [360 + (*cell_lat*(*l, m*)/500000.0 + 180.0) - 0.5] (modulo 360)

disp_lon_index = integer part of [720 + (*cell_lon*(*l, m*)/500000.0 + 360.0) - 0.5] (modulo 720)

`(Req 4.14-12)`

Extract the vegetation class for the cell:

class= *vegetation_class*(*lat_index, lon_index*)

`(Req 4.14-13)`

Step 4.14.4.2 Test for valid data

For each sub-cell $k = 0, \ldots 8$, if either the 11 or 12 micron brightness temperature in the nadir view is invalid, the calculation is abandoned, and the LST is set to -1. The criterion for invalid data is the same as that used for the SST processing, as follows.

Determine the minimum number of pixels required for the cell, for the nadir view.

minpn = 340 * *NADIR_PIXELS_THRESH* + 1.0

`(Req 4.14-14)`

Identify whether the land or 'sea' brightness temperatures are required. Inland lakes are flagged as sea in the current land/sea data-base, so must be treated accordingly.

If *class* = 14 then *sf* = 0 (sea) otherwise *sf* = 1 (land).

If $M(ir12, n; sf, 0, k, l, m) \geq minpn$ and $M(ir11, n; sf, 0, k, l, m) \geq minpn$ proceed to calculate the retrieved LST as below, otherwise set

$T\_land(k, l, m) = -1.0$

(Req 4.14-15)

If the 11 and 12 micron nadir brightness temperatures are valid, Steps 4.14.4.3 to 4.14.4.5 are to be repeated for each sub-cell $k = 0, \ldots 8$ in the cell.

Step 4.14.4.3 Determine day/night flag, satellite elevation and non-linear exponent

If $view\_day(k, l, m)$ = TRUE then

$night = 0$ otherwise $night = 1$

(Req 4.14-16)

A linear interpolation is used to determine the satellite elevation.

$j = across\_track\_mean(1, k, l, m)$

$w = float(j - 6)/50.0 - band(j)$

$i = sub\_cell\_index(k, l, m)$

$sat\_elev = (1.0 - w) \times nadir\_band\_edge\_satellite\_elevation(i, band(j)) +$

$w \times nadir\_band\_edge\_satellite\_elevation(i, band(j) + 1)$

(Req 4.14-17)

Calculate the non-linear exponent:

$\boldsymbol{n} = 1.0 / \cos(\pi \times (90 - sat\_elev) / (m \times 180.0)$

(Req 4.14-18)

Note that m is [L2-AUX10-2] and $\boldsymbol{n}$ is [L2-INT-480].

Step 4.14.4.4 Determine coefficients

$f = 0.001 \times vegetation\_fraction(lat\_index, lon\_index)$

(Req 4.14-19)

Interpolate precipitable water:

$pw00 = precipitable\_water(disp\_lat\_index, disp\_lon\_index)$

$pw01 = precipitable\_water(disp\_lat\_index+1, disp\_lon\_index)$

$pw10 = precipitable\_water(disp\_lat\_index, [disp\_lon\_index+1](\text{modulo } 720))$

$pw11 = precipitable\_water(disp\_lat\_index+1, [disp\_lon\_index+1](\text{modulo } 720))$

(Req 4.14-20)

q = fractional part of $[(cell\_lat(l, m)/500000.0) + 180.0 + 0.5]$

p = fractional part of $[(cell\_lon(l, m)/500000.0) + 360.0 + 0.5]$

$pw = 0.001 \times ((1 - p)(1 - q)pw00 + (1 - p)q \times pw01 + p(1 - q)pw10 + pq \times pw11)$

(Req 4.14-21)

$class = vegetation\_class(lat\_index, lon\_index) - 1$

(Req 4.14-22)

If *class* < 0 or *class* > *NCLASS* - 1 then the index is out of range; the calculation for this sub-cell is abandoned and the nadir field should be set to an exception value of -1.0.

$T\_land(k, l, m) = -1.0$

(Req 4.14-23)

Otherwise

for $k = 0, 2$

$a(k) = f \times coeff(class, 0, night, k) + (1.0 - f) \times coeff(class, 1, night, k)$

(Req 4.14-24)

If (*class* + 1) = 14 this cell is flagged as an inland lake in the vegetation class database. The exponent *n* and the precipitable water correction are not used, and the correct brightness temperature average to be used is that for pixels flagged as sea. Set

n = 1.0.

Otherwise if (*class* + 1) ≠ 14 correct *a*(0) as follows:

$a(0) = a(0) + d \times (cosec(\pi \times sat\_elev / 180.0) - 1.0) \times pw$

(Req 4.14-25)

Note that *d* is [L2-AUX10-1].

Step 4.14.4.5 Calculate the land surface temperature.

Note that the surface flag index *sf* retains the value assigned in Step 4.14.4.2.

If $T^{nadir}_{ir11,sf} > T^{nadir}_{ir12,sf}$ then

$$lst = 100. \times (a(0) + a(1) \times (0.01 \times (T^{nadir}_{ir11,sf} - T^{nadir}_{ir12,sf})))^{**}\mathbf{n}) + (a(1) + a(2)) \times (T^{nadir}_{ir12,sf} - 27315) + 27315$$

(Req 4.14-26)

else

$$lst = 100. \times a(0) + a(1) \times (T^{nadir}_{ir11,sf} - T^{nadir}_{ir12,sf}) + (a(1) + a(2)) \times (T^{nadir}_{ir12,sf} - 27315) + 27315$$

(Req 4.14-27)

Set appropriate bits on AST confidence word *ast_conf*(1; *k*, *l*, *m*):

Set bit 2 if *nadir_day(k, l, m)* from §4.12.3 is TRUE, otherwise clear bit.

(Req 4.14-28)

Set bit 3 if *frwrd_day(k, l, m)* from §4.12.3 is TRUE, otherwise clear bit.

(Req 4.14-29)

Set bits 4 and 5 to the topographic variance flags:

[*ast_conf*(1; *k*, *l*, *m*)](bits 4:5) = *topographic_flag*(*lat_index*, *lon_index*)

(Note that this is a two-bit flag.)

Trap for lst out of range:

If *lst* ≥ 32767.5 then

```
    T_land(k, l, m) = -1                                 (Req 4.14-30)
```

else

```
    T_land(k, l, m) = integer part of (lst + 0.5)        (Req 4.14-31)
```

Step 4.14.5 Calculate 50 km average

For up to nine 17 km cells within the 50 km cell, derive the mean LST [L2-INT-497] for the 50 km cell, and the standard deviation [L2-INT-498] of the 17 km LST values.

$$T\_land(l,m) = \frac{1}{\mu}\sum_k T\_land(k,l,m)$$

(Req 4.14-32)

where the sum is over all values of *k* for which the respective sub-cell LST is valid (i.e. has a positive value), and where μ is the number of valid temperatures. If μ is zero, set the corresponding temperature to –1. To calculate the standard deviation [L2-INT-498] use an expression analogous to Req 4.14-3.

The mean across-track pixel number to be associated with the 30 arc minute LST is [L2-INT-469]

$$sst\_mean\_pixel(1;\, l,\, m) = \frac{1}{\mu}\sum_k across\_track\_mean(1;k,l,m) \text{ if } \mu > 0$$

$$sst\_mean\_pixel(1;\, l,\, m) = -1 \text{ if } \mu = 0$$

(Req 4.14-33)

where the sum is over all $k \in \{0 \le k \le 8\}$ for which corresponding LST *T_land*(*k*, *l*, *m*) is valid (not equal to –1).

Derive the confidence flag word *ast_conf*(1; *l*, *m*) for the 50 km cell indexed by *l*, *m* by ORing together the words for those sub-cells (*k*, *l*, *m*), *k* = 0, ...8, for which a valid temperature was derived.

(Req 4.14-34)

4.15 Module Definition: Spatially Averaged Cloud Parameters (50 km Cell)

4.15.1 Functional Description

This module is to provide physical information on the cloud state additional to the results of the cloud flagging provided by the cloud clearing algorithms. The product is based on the same half-degree cells defined above. The frequency distribution of brightness temperature for the cloudy pixels within the cell is given together with representative parameters and an estimate of the cloud-top temperature. The latter is interpreted as the mean brightness temperature of the coldest 25% of the cloudy pixels in the cell.

For each half-degree cell, information is given for the nadir and forward views separately. The information consists of the number of cloudy and cloud-free pixels falling within the cell, a histogram of the 11 micron brightness temperatures of the cloudy pixels, and various statistical parameters derived from the histogram. The 11 micron channel is used as the basis of the product following the practice of ATSR and ATSR-2.

The product is generated as follows. Two histograms are generated of the frequency distribution of 11 micron brightness temperature, for cloudy pixels over sea and land respectively. The histograms represent the brightness temperature at 0.1 K resolution between 190 K and 290 K. Thus each contains 1000 bins where the first bin contains the number of pixels with temperatures in the range 190.0 to 190.1 K, and the last bin contains the number of pixels with temperatures in the range 289.9 to 290.0 K. The cloud state of each filled pixel falling within the cell is inspected. If it is clear, a count of the number of clear pixels is incremented; if it is cloudy, the 11 micron channel BT is inspected and the count in the appropriate histogram bin is incremented. Note that cosmetic fill pixels are included in the processing.

As each pixel is inspected, a test is made to determine whether its 11 micron BT is lower than the lowest value previously encountered, and if so to store the location of the pixel. Then when the histogram is complete the identity of the minimum pixel will be known, and can be used to extract its channel values.

Once the histogram is complete for a given cell, that is once all the pixels falling within the cell have been inspected, the cloud temperature and coverage results are derived from it. Firstly the total number of cloudy pixels detected is computed by summing the histogram samples. If this total is less than 20 no further derivations are performed. If 20 or more cloudy pixels have been identified and included in the histogram, the mean 11 micron brightness temperature and its standard deviation are calculated from the histogram.

The histogram is searched for the lowest temperature represented by the histogram. This is the temperature corresponding to the first non-zero bin of the histogram. Next, the cloud-top temperature is estimated. The histogram bin containing the 25th percentile is identified; this is the first bin (as the histogram is searched in the direction of ascending temperature) for which the cumulative total of the bins up to and including itself exceeds 25% of the total number of cloudy pixels. The mean temperature represented by the bins up to and including this bin is calculated.

[Note that the cloud top temperature so derived may represent the mean of slightly more than 25% of the cloudy pixels, since the cumulative total including the 25th percentile bin may exceed 25%.]

Finally the percentage cloud cover is calculated from the ratio of cloudy pixels to total pixels.

4.15.2 Interface Definition

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-AUX3-10		AST Cell dime nsion	sl	none	4	1

Table 4-15-1: Input Data Table - Spatially Averaged Cloud Parameters (50 km Cell)

Parameter ID	Variable	Name	Type	Units	Field size	Fields
L2-INT-101	I(ir12, n; i, j)	regridded nadir ir12 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-102	I(ir11, n; i, j)	regridded nadir ir11 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-103	I(ir37, n; i, j)	regridded nadir ir37 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-104	I(v16, n; i, j)	regridded nadir v16 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-105	I(v870, n; i, j)	regridded nadir v870 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-106	I(v670, n; i, j)	regridded nadir v670 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-107	I(v555, n; i, j)	regridded nadir v555 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-111	I(ir12, f; i, j)	regridded forward ir12 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-112	I(ir11, f; i, j)	regridded forward ir11 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-113	I(ir37, f; i, j)	regridded forward ir37 Brightness Temperature	ss	0.01K	2	j = 0, 511
L2-INT-114	I(v16, f; i, j)	regridded forward v16 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-115	I(v870, f; i, j)	regridded forward v870 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-116	I(v670, f; i, j)	regridded forward v670 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-117	I(v555, f; i, j)	regridded forward v555 Reflectance	ss	0.01%	2	j = 0, 511
L2-INT-100	nadir_fill_state(i, j)	nadir fill state indicator	byte	none	1	j = 0, 511
L2-INT-110	frwrd_fill_state(i, j)	frwd fill state indicator	byte	none	1	j = 0, 511
L2-INT-232	nadir_land(i, j)	nadir land/sea flag	ss array	flag	2	j = 0, 511
L2-INT-233	nadir_cloud(i, j)	nadir cloud state flag	ss array	flag	2	j = 0, 511
L2-INT-248	frwrd_land(i, j)	forward land/sea flag	ss array	flag	2	j = 0, 511
L2-INT-249	frwrd_cloud(i, j)	forward cloud state flag	ss array	flag	2	j = 0, 511
L2-INT-160	image_lat(i, j)	image pixel latitude	float	degrees	4	
L2-INT-161	image long(i, j)	image pixel longitude	float	degrees	4	
local	k	histogram bin counter k = 0, 999	sl	none	4	
local	K	25th percentile count	ss	none	2	
	cell	cell number	sl	none	4	
	I_min(ch, v, sf; l, m)	Brightness temperature or reflectance, as appropriate, of channel ch, view v, for cloudy pixel having minimum 11 micron BT over surface type sf.	ss	0.01 K or 0.01%	2	
L2-INT-425	across_track_band	across-track band	ss	none	2	per cell
		land pixel histogram quantities				
L2-INT-428	nadir_clear_land	total of clear land pixels, nadir view	sl	none	4	per cell
L2-INT-429	frwrd_clear_land	total of clear land pixels, forward view	sl	none	4	per cell
L2-INT-430	nadir_cloudy_land	total of cloudy land pixels, nadir view	sl	none	4	per cell
L2-INT-431	frwrd_cloudy_land	total of cloudy land pixels, forward view	sl	none	4	per cell
L2-INT-432	nadir_hist_land(k)	nadir histogram (land cell)	ss	none	2	1000
L2-INT-433	frwrd_hist_land(k)	forward histogram (land cell)	ss	none	2	1000
		nadir output quantities				
L2-INT-435	bt_cloud_top	cloud top temperatuere (over land)	ss	0.01K	2	per cell
L2-INT-436	bt_percent_cloudy	percentage cloudy pixels (over land)	ss	0.01%	2	per cell
		frwrd output quantities				
L2-INT-437	bt_cloud_top	cloud top temperatuere (over land)	ss	0.01K	2	per cell
L2-INT-438	bt_percent_cloudy	percentage cloudy pixels (over land)	ss	0.01%	2	per cell
		sea pixel histogram quantities				
L2-INT-444	nadir_clear_sea	total of clear sea pixels, nadir view	sl	none	4	per cell
L2-INT-445	frwrd_clear_sea	total of clear sea pixels, forward view	sl	none	4	per cell
L2-INT-446	nadir_cloudy_sea	total of cloudy sea pixels, nadir view	sl	none	4	per cell
L2-INT-447	frwrd_cloudy_sea	total of cloudy sea pixels, forward view	sl	none	4	per cell
L2-INT-448	nadir_hist_sea(k)	nadir histogram (sea cell)	ss	none	2	1000
L2-INT-449	frwrd_hist_sea(k)	forward histogram (sea cell)	ss	none	2	1000
		nadir output quantities				
L2-INT-451	bt_cloud_top	cloud top temperatuere (over sea)	ss	0.01K	2	per cell
L2-INT-452	bt_percent_cloudy	percentage cloudy pixels (over sea)	ss	0.01%	2	per cell
		frwrd output quantities				
L2-INT-453	bt_cloud_top	cloud top temperatuere (over sea)	ss	0.01K	2	per cell
L2-INT-454	bt_percent_cloudy	percentage cloudy pixels (over sea)	ss	0.01%	2	per cell

Table 4-15-2: Internal Data Table - Spatially Averaged Cloud Parameters (50 km Cell)

4.15.3 Detailed Structure

Step 4.15.1 Derive histogram for each cell.

Each image row *i* and pixel *j* is used as follows.

Step 4.15.1.1 Identify cell number.

Identify the 50 km cell number *l,m* within which the pixel indexed by *i* and *j* falls exactly as in Section 4.8.3.1.1

If this is the first pixel to fall within a given cell, clear the histogram arrays:

<view>_histogram_<surface>(l, m; k) = 0 for all k = 0, 999

initialise the clear pixel counters

<view>_clear_<surface> = 0,

and initialise associated variables for each channel *ch*:

I_min(ch, v, sf; l, m) = 29000

for each view <*v*> = *nadir*, *frwrd*, and for each surface type <*sf*> = *land*, *sea*.

(Req 4.15-1)

Step 4.15.1.2 Process the image pixel.

Process the image pixel at i, j for both nadir and forward views,

(a) If the pixel is unfilled, do nothing.

(b) If the pixel is over sea, and is clear, increment <view>_clear_sea.

(c) If the pixel is over sea and is cloudy, check the 11 micron brightness temperature. If the 11 micron brightness temperature

$T_{11} = I(ir11, v; i, j)$

is valid, and if $19000 \le T_{11} < 29000$, increment the element of the histogram array <view>_histogram_sea(l, m; k) specified by index

k = integer part of (T_{11}/10 - 19000)

If $T_{11} < I\_min(ir11, v, sf; l, m)$ then set

$I\_min(ch, v, sf; l, m) = I(ch, v; i, j)$

for each channel *ch*.

(d) If the pixel is over land, treat similarly but increment the land counters and histogram arrays <view>_clear_land and <view>_histogram_sea(l, m; k) in place of the corresponding sea variables.

(Req 4.15-2)

Step 4.15.2 Process histograms.

In this step the following notation is used.

$N_{v,sf}^{cloud}(l,m)$	<view>_cloudy_<surface>
$F_{v,sf}(l,m;k)$	<view>_hist_<surface>(l, m; k) where <view> = nadir | frwrd and <surface> = sea | land
$T_{v,sf}^{ct}(l,m)$	[bt_cloud_top](v, sf)

When all four histograms are complete, find the number of cloudy pixels in each. For each view v = n, f and for each surface type sf:

$$N_{v,sf}^{cloud}(l,m) = \sum_{k=0}^{999} F_{v,sf}(l,m;k)$$ (Req 4.15-3)

If the number of cloudy pixels found is less than 20, proceed to the next cell. Otherwise proceed as follows:

Calculate the position of the 25th percentile

$$K = N_{v,sf}^{cloud}(l,m)/4$$ (Req 4.15-4)

and find the index k(such that

$$\sum_{k=0}^{k'-1} F_{v,sf}(l,m;k) < K \le \sum_{k=0}^{k'} F_{v,sf}(l,m;k)$$ (Req 4.15-5)

Then the cloud-top temperature is given by

$$T_{v,sf}^{ct}(l,m) = 19000 + 10 \cdot \sum_{k=0}^{k'} (k+0.5) \cdot F_{v,sf}(l,m;k) \Big/ \sum_{k=0}^{k'} F_{v,sf}(l,m;k)$$

(Req 4.15-6)

and the percentage of cloudy pixels for each view and surface type is given by

$$\text{bt\_percent\_cloudy} = 10000 \cdot \frac{N_{v,sf}^{cloud}(l,m)}{N_{v,sf}^{cell}(l,m) + <view>\_clear\_<surface>(l,m)}$$

(Req 4.15-7)

4.16 Module Definition: Output AST Product

4.16.1 Functional Description

The AST product is written to the output medium. First the MPH, and SPH are written, then the Measurement data sets.

4.16.2 Interface Definition

See IODD Tables and Internal Parameter List

4.16.3 Detailed Structure

Step 4.16.1 AST MDS #1: Sea Cell SST Record, 50 km cell

Record identified by cell indices *l*, *m*.

First convert the cell UTC [L2-INT-20] to transport format for output, using the ESA CFI subroutine `pl_pmjd`.

```
mjdp[0]/(1) = utc(l, m)
mjdp[1]/(2) = 0.0 (dummy value since output not required)
status = pl_pmjd(mjdt, mjdp, utce, dut1e)
```

[AST-MDS1-1](*l*, *m*) = [mjdt[0:2]/(1:3)](*l*, *m*) (Req 4.16.1-1)

[AST-MDS1-2](*l*, *m*) = -1 if (*N_sea(n; l, m)* = 0 and *N_sea(f; l, m)* = 0)
= 0 otherwise (Req 4.16.1-2)

[AST-MDS1-3](*l*, *m*) = (3 zero bytes) (Req 4.16.1-3)

[AST-MDS1-4](*l*, *m*) = *cell_lat(l, m)* [L2-INT-77] (Req 4.16.1-4)

[AST-MDS1-5](*l*, *m*) = *cell_long(l, m)* [L2-INT-78] (Req 4.16.1-5)

[AST-MDS1-6](*l*, *m*) = *sst_mean_pixel*(0; *l, m)* [L2-INT-469] (Req 4.16.1-6)

[AST-MDS1-7](*l*, *m*) = *T_nadir(l, m)* (Req 4.16.1-7)

[AST-MDS1-8](*l*, *m*) = *σ_nadir(ASST; l, m)* (Req 4.16.1-8)

[AST-MDS1-9](*l*, *m*) = the smaller of $\tilde{M}$ *(ir11, n; 0, 0, l, m),*
$\tilde{M}$ *(ir12, n; 0, 0, l, m)* (Req 4.16.1-9)

[AST-MDS1-10](*l*, *m*) = *T_dual(l, m)* (Req 4.16.1-10)

[AST-MDS1-11](*l*, *m*) = *σ_dual(ASST; l, m)* (Req 4.16.1-11)

[AST-MDS4-12](*cell*) = the smallest of $\tilde{M}$ *(ir11, n; 0, 0, l, m),*
$\tilde{M}$ *(ir12, n; 0, 0, l, m),* $\tilde{M}$ *(ir11, f; 0, 0, l, m),*
$\tilde{M}$ *(ir12, f; 0, 0, l, m)* (Req 4.16.4-12)

[AST-MDS1-13](*l*, *m*) = *ast_conf*(0; *l, m)* (Req 4.16.1-13)

Averaged cloud parameters (ACLOUD), nadir view:

[AST-MDS1-23](*l*, *m*) = [L2-INT-451]*(l, m)* (Req 4.16.1-23)

[AST-MDS1-24](*l*, *m*) = [L2-INT-452]*(l, m)* (Req 4.16.1-24)

Averaged cloud parameters (ACLOUD), forward view:

[AST-MDS1-32](*l*, *m*) = [L2-INT-453]*(l, m)* (Req 4.16.1-32)

[AST-MDS1-33](*l*, *m*) = [L2-INT-454]*(l, m)* (Req 4.16.1-33)

Step 4.16.2 AST MDS #2: Sea Cell SST record, 17 km cell:

Record identified by (*k*, *l*, *m*)

First convert the cell UTC [L2-INT-21] to transport format for output, using the ESA CFI subroutine `pl_pmjd`.

```
mjdp[0]/(1) = utc(k, l, m)
mjdp[1]/(2) = 0.0 (dummy value since output not required)
status = pl_pmjd(mjdt, mjdp, utce, dut1e)
```

[AST-MDS2-1](*k*, *l*, *m*) = [mjdt[0:2]/(1:3)](*k*, *l*, *m*) (Req 4.16.2-1)

[AST-MDS2-2](*k*, *l*, *m*) = -1 if (*N_sea(n; k, l, m)* = 0 and *N_sea(f; k, l, m)* = 0)
= 0 otherwise (Req 4.16.2-2)

[AST-MDS2-3] (*k*, *l*, *m*) = (3 zero bytes) (Req 4.16.2-3)

[AST-MDS2-4](*k*, *l*, *m*) = *sub_cell_lat(k, l, m)* [L2-INT-62] (Req 4.16.2-4)

[AST-MDS2-5](*k*, *l*, *m*) = *sub_cell_long(k, l, m)* [L2-INT-63] (Req 4.16.2-5)

[AST-MDS2-6](*k*, *l*, *m*) = *across_track_mean*(0; *k*, *l*, *m)* [L2-INT-459] (Req 4.16.2-6)

[AST-MDS2-7](*k*, *l*, *m*) = *T_nadir(k, l, m)* (Req 4.16.2-7)

[AST-MDS2-8](*k*, *l*, *m*) = the smaller of *M(ir11, n; 0, 0, k, l, m),*
M(ir12, n; 0, 0, k, l, m) (Req 4.16.2-8)

[AST-MDS2-9](*k*, *l*, *m*) = *T_dual(k, l, m)* (Req 4.16.2-9)

[AST-MDS2-10](*k*, *l*, *m*) = the smallest of *M(ir11, n; 0, 0, k, l, m),*
M(ir12, n; 0, 0, k, cell), M(ir11, f; 0, 0, k, l, m),
M(ir12, f; 0, 0, k, l, m) (Req 4.16.2-10)

[AST-MDS2-11](*k*, *l*, *m*) = *ast_conf(0; k, l, m)* (Req 4.16.2-11)

Step 4.16.3 AST MDS #3: Sea Cell SST record, 10 arc minute cell:

Record identified by (k, cell)

First convert the cell UTC [L2-INT-31] to transport format for output, using the ESA CFI subroutine `pl_pmjd`.

```
mjdp[0]/(1) = utc(k, cell)
mjdp[1]/(2) = 0.0 (dummy value since output not required)
status = pl_pmjd(mjdt, mjdp, utce, dut1e)
```

[AST-MDS3-1](*k*, *cell*) = [mjdt[0:2]/(1:3)](*k*, *cell*) (Req 4.16.3-1)

[AST-MDS3-2](*k*, *cell*) = -1 if (*N_sea(n; k, cell)* = 0 and *N_sea(f;k, cell)* = 0)
= 0 otherwise (Req 4.16.3-2)

[AST-MDS3-3](*k*, *cell*) = (3 zero bytes) (Req 4.16.3-3)

[AST-MDS3-4](*k*, *cell*) = *sub_cell_lat(k, cell)* [L2-INT-32] (Req 4.16.3-4)

[AST-MDS3-5](*k*, *cell*) = *sub_cell_long(k, cell)* [L2-INT-33] (Req 4.16.3-5)

[AST-MDS3-6](*k*, *cell*) = *across_track_mean*(0; *k*, *cell)* [L2-INT-359] (Req 4.16.3-6)

[AST-MDS3-7](*k*, *cell*) = *T_nadir(k, cell)* [L2-INT-54] (Req 4.16.3-7)

[AST-MDS3-8](*k*, *cell*) = the smaller of *M(ir11, n; 0, 0, k, cell),*
M(ir12, n; 0, 0, k, cell) (Req 4.16.3-8)

[AST-MDS3-9](*k, cell*) = *T_dual(k, cell)* [L2-INT-56] (Req 4.16.3-9)

[AST-MDS3-10](*k, cell*) = the smallest of *M(ir11, n; 0, 0, k, cell), M(ir12, n; 0, 0, k, cell), M(ir11, f; 0, 0, k, cell), M(ir12, f; 0, 0, k, cell)* (Req 4.16.3-10)

[AST-MDS3-11](*k, cell*) = *ast_conf(0; k, cell)* (Req 4.16.3-11)

Step 4.16.4 AST MDS #4: Sea Cell SST Record, 30 arc minute cell

Record identified by cell number *cell*.

First convert the cell UTC [L2-INT-30] to transport format for output, using the ESA CFI subroutine `pl_pmjd`.

```
mjdp[0]/(1) = utc(cell)
mjdp[1]/(2) = 0.0 (dummy value since output not required)
status = pl_pmjd(mjdt, mjdp, utce, dut1e)
```

[AST-MDS4-1](*cell*) = [mjdt[0:2]/(1:3)](*cell*) (Req 4.16.4-1)

[AST-MDS4-2](*cell*) = -1 if (*N_sea(n; cell)* = 0 and *N_sea(f; cell)* = 0)
= 0 otherwise (Req 4.16.4-2)

[AST-MDS4-3](*cell*) = (3 zero bytes) (Req 4.16.4-3)

[AST-MDS4-4](*cell*) = *cell_lat(cell)* [L2-INT-47] (Req 4.16.4-4)

[AST-MDS4-5](*cell*) = *cell_long(cell)* [L2-INT-48] (Req 4.16.4-5)

[AST-MDS4-6](*cell*) = sst_mean_pixel(0; cell) [L2-INT-369] (Req 4.16.4-6)

[AST-MDS4-7](*cell*) = *T_nadir(cell)* (Req 4.16.4-7)

[AST-MDS4-8](*cell*) = *σ_nadir(ASST; cell)* (Req 4.16.4-8)

[AST-MDS4-9](*cell*) = the smaller of $\tilde{M}$ *(ir11, n; 0, 0, cell),* $\tilde{M}$ *(ir12, n; 0, 0, cell)* (Req 4.16.4-9)

[AST-MDS4-10](*cell*) = *T_dual(cell)* (Req 4.16.4-10)

[AST-MDS4-11](*cell*) = *σ_dual(ASST; cell)* (Req 4.16.4-11)

[AST-MDS4-12](*cell*) = the smallest of $\tilde{M}$ *(ir11, n; 0, 0, cell),* $\tilde{M}$ *(ir12, n; 0, 0, cell),* $\tilde{M}$ *(ir11, f; 0, 0, cell),* $\tilde{M}$ *(ir12, f; 0, 0, cell)* (Req 4.16.4-12)

[AST-MDS4-13](*cell*) = *ast_conf(0; cell)* (Req 4.16.4-13)

Averaged cloud parameters (ACLOUD), nadir view:

[AST-MDS4-23](*cell*) = [L2-INT-351]*(cell)* (Req 4.16.4-23)

[AST-MDS4-24](*cell*) = [L2-INT-352]*(cell)* (Req 4.16.4-24)

Averaged cloud parameters (ACLOUD), forward view:

[AST-MDS4-32](*cell*) = [L2-INT-353]*(cell)* (Req 4.16.4-32)

[AST-MDS4-33](*cell*) = [L2-INT-354]*(cell)* (Req 4.16.4-33)

Step 4.16.5 AST MDS #5: Land Cell LST/NDVI Record, 50 km cell

Record identified by cell indices *l, m*.

[AST-MDS5-1](*l, m*) = [AST-MDS1-1](*l, m*) (Req 4.16.5-1)

[AST-MDS5-2](*l, m*) = -1 if (*N_land(n; l, m)* = 0 and *N_land(f; l, m)* = 0)
= 0 otherwise (Req 4.16.5-2)

[AST-MDS5-3](*l, m*) = (3 zero bytes) (Req 4.16.5-3)

[AST-MDS5-4](*l, m*) = *cell_lat(l, m)* [L2-INT-77] (Req 4.16.5-4)

[AST-MDS5-5](*l, m*) = *cell_long(l, m)* [L2-INT-78] (Req 4.16.5-5)

[AST-MDS5-6](*l, m*) = *sst_mean_pixel(*1; *l, m)* [L2-INT-469] (Req 4.16.5-6.1)

[AST-MDS5-7](*l, m*) = *T_land(l, m)* [L2-INT-497] (Req 4.16.5-7.1)

[AST-MDS5-8](*l, m*) = σ*_land(l, m)* [L2-INT-498] (Req 4.16.5-8.1)

[AST-MDS5-9](*l, m*) = the smaller of $\tilde{M}(ir11, n; sf, 0, l, m)$,
$\tilde{M}(ir12, n; sf, 0, l, m)$ (Req 4.16.5-9.1)

where *sf* has the value assigned to it in step 4.14.4.2.

[AST-MDS5-10](*l, m*) = <NDVI>(*l, m*) [L2-INT-96] (Req 4.16.5-10)

[AST-MDS5-11](*l, m*) = σ(NDVI: *l, m*) [L2-INT-97] (Req 4.16.5-11)

[AST-MDS5-12](*l, m*) = *N0(l, m)* [L2-INT-98] (Req 4.16.5-12)

[AST-MDS5-13](*l, m*) = *ast_conf(1; l, m)* (Req 4.16.5-13)

Averaged cloud parameters (ACLOUD), nadir view:

[AST-MDS5-23](*l, m*) = [L2-INT-435]*(l, m)* (Req 4.16.5-23)

[AST-MDS5-24](*l, m*) = [L2-INT-436]*(l, m)* (Req 4.16.5-24)

Averaged cloud parameters (ACLOUD), forward view:

[AST-MDS5-32](*l, m*) = [L2-INT-437]*(l, m)* (Req 4.16.5-32)

[AST-MDS5-33](*l, m*) = [L2-INT-438]*(l, m)* (Req 4.16.5-33)

Step 4.16.6 AST MDS #6: Land Cell LST/NDVI record, 17 km cell:

Record identified by (*k, l, m*)

[AST-MDS6-1](*k, l, m*) = [AST-MDS2-1](*k, l, m*) (Req 4.16.6-1)

[AST-MDS6-2](*k, l, m*) = -1 if (*N_land(n; k, l, m)* = 0 and *N_land(f; k, l, m)* = 0)
= 0 otherwise (Req 4.16.6-2)

[AST-MDS6-3](*k, l, m*) = (3 zero bytes) (Req 4.16.6-3)

[AST-MDS6-4](*k, l, m*) = *sub_cell_lat(k, l, m)* [L2-INT-62] (Req 4.16.6-4)

[AST-MDS6-5](*k, l, m*) = *sub_cell_long(k, l, m)* [L2-INT-63] (Req 4.16.6-5)

[AST-MDS6-6](*k, l, m*) = *across_track_mean*(1; *k, l, m)* [L2-INT-459] (Req 4.16.6-6)

[AST-MDS6-7](*k, l, m*) = *T_land(k, l, m)* [L2-INT-496] (Req 4.16.6-7.1)

[AST-MDS6-8](*k, l, m*) = the smaller of *M(ir11, n; sf, 0, k, l, m),*
M(ir12, n; sf, 0, k, l, m) (Req 4.16.6-8.1)

where *sf* has the value assigned to it in step 4.14.4.2.

[AST-MDS6-9](*k, l, m*) = *NDVI*(*k, l, m*) [L2-INT-95] (Req 4.16.6-9)

[AST-MDS6-10](*k, l, m*) = N1(*k, l, m*) [L2-INT-99] (Req 4.16.6-10)

[AST-MDS6-11](*k, l, m*) = *ast_conf(1; k, l, m)* (Req 4.16.6-11)

Step 4.16.7 AST MDS #7: Land Cell LST/NDVI record, 10 arcminute cell:

Record identified by *(k, cell*)

[AST-MDS7-1](*k, cell*) = [AST-MDS3-1](*k, cell*) (Req 4.16.7-1)

[AST-MDS7-2](*k, cell*) = -1 if (*N_land(n; k, cell)* = 0 and *N_land(f;k, cell)* = 0)
= 0 otherwise (Req 4.16.7-2)

[AST-MDS7-3](*k, cell*) = (3 zero bytes) (Req 4.16.7-3)

[AST-MDS7-4](*k, cell*) = *sub_cell_lat(k, cell)* [L2-INT-32] (Req 4.16.7-4)

[AST-MDS7-5](*k, cell*) = *sub_cell_long(k, cell)* [L2-INT-33] (Req 4.16.7-5)

[AST-MDS7-6](*k, cell*) = *across_track_mean*(1; *k, cell)* [L2-INT-359] (Req 4.16.7-6)

[AST-MDS7-7](*k, cell*) = *T_land(k, cell)* [L2-INT-492] (Req 4.16.7-7.1)

[AST-MDS7-8](*k, cell*) = the smaller of *M(ir11, n; sf, 0, k, cell),*
M(ir12, n; sf, 0, k, cell) (Req 4.16.7-8.1)

where *sf* has the value assigned to it in step 4.10.4.2.

[AST-MDS7-9]*(k, cell) = NDVI(k, cell)* [L2-INT-90] (Req 4.16.7-9)

[AST-MDS7-10]*(k, cell) = N1(k, cell)* [L2-INT-94] (Req 4.16.7-10)

[AST-MDS7-11](*k, cell*) = *ast_conf(1; k, cell)* (Req 4.16.7-11)

Step 4.16.8 AST MDS #8: Land Cell LST/NDVI Record, 30 arc minute cell

Record identified by cell number *cell*.

[AST-MDS8-1](*cell*) = [AST-MDS4-1](*cell*) (Req 4.16.8-1)

[AST-MDS8-2](*cell*) = -1 if (*N_land(n; cell)* = 0 and *N_land(f; cell)* = 0)
= 0 otherwise (Req 4.16.8-2)

[AST-MDS8-3](*cell*) = (3 zero bytes) (Req 4.16.8-3)

[AST-MDS8-4](*cell*) = *cell_lat(cell)* [L2-INT-47] (Req 4.16.8-4)

[AST-MDS8-5](*cell*) = *cell_long(cell)* [L2-INT-48] (Req 4.16.8-5)

[AST-MDS8-6](*cell*) = sst_mean_pixel(1; cell) [L2-INT-369] (Req 4.16.8-6.1)

[AST-MDS8-7](*cell*) = *T_land(cell)* [L2-INT-493] (Req 4.16.8-7.1)

[AST-MDS8-8](*cell*) = σ*_land(cell)* [L2-INT-494] (Req 4.16.8-8.1)

[AST-MDS8-9](*cell*) = the smaller of $\tilde{M}$ *(ir11, n; sf, 0, cell),*
$\tilde{M}$ *(ir12, n; sf, 0, cell)* (Req 4.16.8-9.1)

where *sf* has the value assigned to it in step 4.10.4.2.

[AST-MDS8-10](*cell*) = <NDVI>(*cell*) [L2-INT-91] (Req 4.16.8-10)

[AST-MDS8-11](*cell*) = σ(NDVI: *cell*) [L2-INT-92] (Req 4.16.8-11)

[AST-MDS8-12](*cell*) = *N0(cell)* [L2-INT-93] (Req 4.16.8-12)

[AST-MDS8-13](*cell*) = *ast_conf(1; cell)* (Req 4.16.8-13)

Averaged cloud parameters (ACLOUD), nadir view:

[AST-MDS8-23](*cell*) = [L2-INT-335]*(cell)* (Req 4.16.8-23)

[AST-MDS8-24](*cell*) = [L2-INT-336]*(cell)* (Req 4.16.8-24)

Averaged cloud parameters (ACLOUD), forward view:

[AST-MDS8-32](*cell*) = [L2-INT-337]*(cell)* (Req 4.16.8-32)

[AST-MDS8-33](*cell*) = [L2-INT-338]*(cell)* (Req 4.16.8-33)

Step 4.16.9 AST MDS#9: Land Cell BT/TOA Record, 50 km cell

Record identified by cell number *l*, *m*.

[AST-MDS9-1](*l*, *m*) = [AST-MDS1-1](*l*, *m*) (Req 4.16.9-1)

[AST-MDS9-2](*l*, *m*) = -1 if (*N_land(n; l, m)* = 0 and *N_land(f; l, m)* = 0)
= 0 otherwise (Req 4.16.9-2)

[AST-MDS9-3](*l*, *m*) = (3 zero bytes) (Req 4.16.9-3)

[AST-MDS9-4](*l*, *m*) = *cell_lat(l, m)* [L2-INT-77] (Req 4.16.9-4)

[AST-MDS9-5](*l*, *m*) = *cell_long(l, m)* [L2-INT-78] (Req 4.16.9-5)

[AST-MDS9-6](*l*, *m*) = *across_track_mean*(1; *l*, *m)* [L2-INT-468] (Req 4.16.9-6)

[AST-MDS9-7](*l*, *m*) = *N_total(n; l, m)* [L2-INT-408] (Req 4.16.9-7)

[AST-MDS9-8](*l*, *m*) = *N_land(n; l, m)* [L2-INT-406] (Req 4.16.9-8)

[AST-MDS9-9](*l*, *m*) = *pcl(n; l, m)* [L2-INT-410] (Req 4.16.9-9)

[AST-MDS9-10](*l*, *m*) = Topographic latitude correction (Req 4.16.9-10)

[AST-MDS9-11](*l*, *m*) = Topographic longitude correction (Req 4.16.9-11)

Clear pixels, nadir.

[AST-MDS9-<12 + 2.(ch - 1)>](*l*, *m*) = $\tilde{A}$ (ch, n; 1, 0, *l*, *m*) (Req 4.16.9-12)

[AST-MDS9-<13 + 2.(ch - 1)>](*l*, *m*) = σ(ch, n; 1, 0, *l*, *m*) (Req 4.16.9-13)

Cloudy pixels, nadir.

[AST-MDS9-<26 + 2.(ch - 1)>](*l*, *m*) = $\tilde{A}$(ch, n; 1, 1, *l*, *m*) (Req 4.16.9-14)

[AST-MDS9-<27 + 2.(ch - 1)>](*l*, *m*) = σ(ch, n; 1, 1, *l*, *m*) (Req 4.16.9-15)

[AST-MDS9-40](*l*, *m*) = *PFF(n, land; l, m)* [L2-INT-81] (Req 4.16.9-16)

Forward view:

[AST-MDS9-41](*l*, *m*) = *N_total(f; l, m)* [L2-INT-418] (Req 4.16.9-17)

[AST-MDS9-42](*l*, *m*) = *N_land(f; l, m)* [L2-INT-416] (Req 4.16.9-18)

[AST-MDS9-43](*l*, *m*) = *pcl(f; l, m)* [L2-INT-420] (Req 4.16.9-19)

[AST-MDS9-44](*l*, *m*) = Topographic latitude correction (Req 4.16.9-20)

[AST-MDS9-45](*l*, *m*) = Topographic longitude correction (Req 4.16.9-21)

Clear pixels, forward.

[AST-MDS9-<46 + 2.(ch - 1)>](*l*, *m*) = $\tilde{A}$(ch,f; 1, 0, *l*, *m*) (Req 4.16.9-22)

[AST-MDS9-<47 + 2.(ch - 1)>](*l*, *m*) = σ(ch, f; 1, 0, *l*, *m*) (Req 4.16.9-23)

Cloudy pixels, forward.

[AST-MDS9-<60 + 2.(ch - 1)>](*l*, *m*) = $\tilde{A}$(ch, f; 1, 1, *l*, *m*) (Req 4.16.9-24)

[AST-MDS9-<61 + 2.(ch - 1)>](*l*, *m*) = σ(ch, f; 1, 1, *l*, *m*) (Req 4.16.9-25)

[AST-MDS9-74](*l*, *m*) = *PFF(f, land; l, m)* [L2-INT-81] (Req 4.16.9-26)

[AST-MDS9-75](*l*, *m*) = N0(*l*, *m*) (Req 4.16.9-27)

[AST-MDS9-76](*l*, *m*) = (ACLOUD) Percentage of pixels over <surface type>
(Req 4.16.9-28)

Averaged cloud parameters (ACLOUD), nadir view:

[AST-MDS9-77](*l*, *m*) = *I_min(ir11, n, land; l, m)* (Req 4.16.9-29)

[AST-MDS9-78](*l*, *m*) = *I_min(ir12, n, land; l, m)* (Req 4.16.9-30)

[AST-MDS9-79](*l*, *m*) = *I_min(ir37, n, land; l, m)* (Req 4.16.9-31)

[AST-MDS9-80](*l*, *m*) = *I_min(v16, n, land; l, m)* (Req 4.16.9-32)

[AST-MDS9-81](*l*, *m*) = *I_min(v870, n, land; l, m)* (Req 4.16.9-33)

[AST-MDS9-82](*l*, *m*) = *I_min(v670, n, land; l, m)* (Req 4.16.9-34)

[AST-MDS9-83](*l*, *m*) = *I_min(v555, n, land; l, m)* (Req 4.16.9-35)

Averaged cloud parameters (ACLOUD), forward view:

[AST-MDS9-84](*l*, *m*) = *I_min(ir11, f, land; l, m)* (Req 4.16.9-36)

[AST-MDS9-85](*l*, *m*) = *I_min(ir12, f, land; l, m)* (Req 4.16.9-37)

[AST-MDS9-86](*l*, *m*) = *I_min(ir37, f, land; l, m)* (Req 4.16.9-38)

[AST-MDS9-87](*l*, *m*) = *I_min(v16, f, land; l, m)* (Req 4.16.9-39)

[AST-MDS9-88](*l*, *m*) = *I_min(v870, f, land; l, m)* (Req 4.16.9-40)

[AST-MDS9-89](*l*, *m*) = *I_min(v670, f, land; l, m)* (Req 4.16.9-41)

[AST-MDS9-90](*l*, *m*) = *I_min(v555, f, land; l, m)* (Req 4.16.9-42)

Step 4.16.10 AST MDS#10: Land Cell BT/TOA Record, 17 km cell

Record identified by (*k*, *l*, *m*)

Clear pixels, nadir.

[AST-MDS10-1](*k*, *l*, *m*) = [AST-MDS2-1](*k*, *l*, *m*) (Req 4.16.10-1)

[AST-MDS10-2](*k*, *l*, *m*) = -1 if (*N_land(n; k, l, m)* = 0 and *N_land(f; k, l, m)* = 0)
= 0 otherwise (Req 4.16.10-2)

[AST-MDS10-3](*k*, *l*, *m*) = (3 zero bytes) (Req 4.16.10-3)

[AST-MDS10-4](*k*, *l*, *m*) = *sub_cell_lat(k, l, m)* [L2-INT-62] (Req 4.16.10-4)

[AST-MDS10-5](*k*, *l*, *m*) = *sub_cell_long(k, l, m)* [L2-INT-63] (Req 4.16.10-5)

[AST-MDS10-6](*k*, *l*, *m*) = *across_track_mean*(1; *k, l, m)* [L2-INT-459] (Req 4.16.10-6)

[AST-MDS10-7](*k*, *l*, *m*) = *N_total(n; k, l, m)* [L2-INT-403] (Req 4.16.10-7)

[AST-MDS10-8](*k*, *l*, *m*) = *N_land(n; k, l, m)* [L2-INT-401] (Req 4.16.10-8)

[AST-MDS10-9](*k*, *l*, *m*) = *pcl(n; k, l, m)* [L2-INT-405] (Req 4.16.10-9)

[AST-MDS10-10](*l*, *m*) = Topographic latitude correction (Req 4.16.10-10)

[AST-MDS10-11](*l*, *m*) = Topographic longitude correction (Req 4.16.10-11)

[AST-MDS10-<12 + (ch - 1)>](*k*, *l*, *m*) = A(ch, n; 1 0, *k*, *l*, *m*) (Req 4.16.10-12)

Cloudy pixels, nadir.

[AST-MDS10-<19 + (ch - 1)>](*k*, *l*, *m*) = A(ch, n; 1, 1, *k*, *l*, *m*) (Req 4.16.10-13)

[AST-MDS10-26](*k*, *l*, *m*) = *PFF(n, land; k, l, m)* [L2-INT-82] (Req 4.16.10-14)

Forward view:

[AST-MDS10-27](*k*, *l*, *m*) = *N_total(f; k, l, m)* [L2-INT-413] (Req 4.16.10-15)

[AST-MDS10-28](*k*, *l*, *m*) = *N_land(f; k, l, m)* [L2-INT-411] (Req 4.16.10-16)

[AST-MDS10-29](*k*, *l*, *m*) = *pcl(f; k, l, m)* [L2-INT-415] (Req 4.16.10-17)

[AST-MDS10-30](*l*, *m*) = Topographic latitude correction (Req 4.16.10-18)

[AST-MDS10-31](*l*, *m*) = Topographic longitude correction (Req 4.16.10-19)

Clear pixels, forward.

[AST-MDS10-<32 + (ch - 1)>](*k*, *l*, *m*) = A(ch, f; 1, 0, *k*, *l*, *m*) (Req 4.16.10-20)

Cloudy pixels, forward.

[AST-MDS10-<39 + (ch - 1)>](*k*, *l*, *m*) = A(ch, f; 1, 1, *k*, *l*, *m*) (Req 4.16.10-21)

[AST-MDS10-46](*k*, *l*, *m*) = *PFF(f, land; k, l, m)* [L2-INT-82] (Req 4.16.10-22)

Step 4.16.11 AST MDS#11: Land Cell BT/TOA Record, 10 arc minute cell

Record identified by *(k, cell)*

[AST-MDS11-1](*k*, *cell*) = [AST-MDS3-1](*k*, *cell*) (Req 4.16.11-1)

[AST-MDS11-2](*k*, *cell*) = -1 if (*N_land(n; k, cell)* = 0 and *N_land(f;k, cell)* = 0)
= 0 otherwise (Req 4.16.11-2)

[AST-MDS11-3](*k*, *cell*) = (3 zero bytes) (Req 4.16.11-3)

[AST-MDS11-4](*k*, *cell*) = *sub_cell_lat(k, cell)* [L2-INT-32] (Req 4.16.11-4)

[AST-MDS11-5](*k*, *cell*) = *sub_cell_long(k, cell)* [L2-INT-33] (Req 4.16.11-5)

[AST-MDS11-6](*k*, *cell*) = *across_track_mean(*1; *k, cell)* [L2-INT-359] (Req 4.16.11-6)

[AST-MDS11-7](*k*, *cell*) = *N_total(n; k, cell)* [L2-INT-303] (Req 4.16.11-7)

[AST-MDS11-8](*k*, *cell*) = *N_land(n; k, cell)* [L2-INT-301] (Req 4.16.11-8)

[AST-MDS11-9](*k*, *cell*) = *pcl(n; k, cell)* [L2-INT-305] (Req 4.16.11-9)

[AST-MDS11-10](*k*, *cell*) = Topographic latitude correction (Req 4.16.11-10)

[AST-MDS11-11](*k*, *cell*) = Topographic longitude correction (Req 4.16.11-11)

Clear pixels, nadir.

[AST-MDS11-<12 + (ch - 1)>](*k*, *cell*) = *A(ch, n;* 1, 0, *k*, *cell)* (Req 4.16.11-12)

Cloudy pixels, nadir.

[AST-MDS11-<19 + (ch - 1)>](*k*, *cell*) = *A(ch, n;* 1, 1, *k*, *cell)* (Req 4.16.11-13)

[AST-MDS11-26](*k*, *cell*) = *PFF(n, land; k, cell)* [L2-INT-52] (Req 4.16.11-14)

Forward view:

[AST-MDS11-27](*k*, *cell*) = *N_total(f; k, cell)* [L2-INT-313] (Req 4.16.11-15)

[AST-MDS11-28](*k*, *cell*) = *N_land(f; k, cell)* [L2-INT-311] (Req 4.16.11-16)

[AST-MDS11-29](*k*, *cell*)= *pcl(f; k, cell)* [L2-INT-315] (Req 4.16.11-17)

[AST-MDS11-30](*k*, *cell*) = Topographic latitude correction (Req 4.16.11-18)

[AST-MDS11-31](*k*, *cell*) = Topographic longitude correction (Req 4.16.11-19)

Clear pixels, forward.

[AST-MDS11-<32 + (ch - 1)>](*k*, *cell*) = *A(ch, f;* 1, 0, *k*, *cell)* (Req 4.16.11-20)

Cloudy pixels, forward.

[AST-MDS11-<39 + (ch - 1)>](*k, cell*) = *A(ch, f;* 1 1, *k, cell)* (Req 4.16.11-21)

[AST-MDS11-46](*k, cell*) = *PFF(f, land; k, cell)* [L2-INT-52] (Req 4.16.11-22)

Step 4.16.12 AST MDS#12: Land Cell BT/TOA Record, 30 arc minute cell

Record identified by cell number *cell*.

[AST-MDS12-1](*cell*) = [AST-MDS4-1](*cell*) (Req 4.16.12-1)

[AST-MDS12-2](*cell*) = -1 if (*N_land(n; cell)* = 0 and *N_land(f; cell)* = 0)
= 0 otherwise (Req 4.16.12-2)

[AST-MDS12-3](*cell*) = (3 zero bytes) (Req 4.16.12-3)

[AST-MDS12-4](*cell*) = *cell_lat(cell)* [L2-INT-47] (Req 4.16.12-4)

[AST-MDS12-5](*cell*) = *cell_long(cell)* [L2-INT-48] (Req 4.16.12-5)

[AST-MDS12-6](*cell*) = *across_track_mean(*1; *cell)* [L2-INT-368] (Req 4.16.12-6)

[AST-MDS12-7](*cell*) = *N_total(n; cell)* [L2-INT-308] (Req 4.16.12-7)

[AST-MDS12-8](*cell*) = *N_land(n; cell)* [L2-INT-306] (Req 4.16.12-8)

[AST-MDS12-9](*cell*) = *pcl(n; cell)* [L2-INT-310] (Req 4.16.12-9)

[AST-MDS12-10](*cell*) = Topographic latitude correction (Req 4.16.12-10)

[AST-MDS12-11](*cell*) = Topographic longitude correction (Req 4.16.12-11)

Clear pixels, nadir.

[AST-MDS12-<12 + 2.(ch - 1)>](*cell*) = $\tilde{A}$ (*ch, n*; 1, 0, cell) (Req 4.16.12-12)

[AST-MDS12-<13 + 2.(ch - 1)>](*cell*) = σ(*ch, n;* 1, 0, *cell*) (Req 4.16.12-13)

Cloudy pixels, nadir.

[AST-MDS12-<26 + 2.(ch - 1)>](*cell*) = $\tilde{A}$ (*ch, n;* 1, 1, *cell*) (Req 4.16.12-14)

[AST-MDS12-<27 + 2.(ch - 1)>](*cell*) = σ(*ch, n;* 1, 1, *cell*) (Req 4.16.12-15)

[AST-MDS12-40](*cell*) = *PFF(n, land; cell)* [L2-INT-51] (Req 4.16.12-16)

Forward view:

[AST-MDS12-41](*cell*) = *N_total(f; cell)* [L2-INT-318] (Req 4.16.12-17)

[AST-MDS12-42](*cell*) = *N_land(f; cell)* [L2-INT-316] (Req 4.16.12-18)

[AST-MDS12-43](*cell*) = *pcl(f; cell)* [L2-INT-320] (Req 4.16.12-19)

[AST-MDS12-44](*cell*) = Topographic latitude correction (Req 4.16.12-20)

[AST-MDS12-45](*cell*) = Topographic longitude correction (Req 4.16.12-21)

Clear pixels, forward.

[AST-MDS12-<46 + 2.(ch - 1)>](*cell*) = $\tilde{A}$ (*ch,f;* 1, 0, *cell*) (Req 4.16.12-22)

[AST-MDS12-<47 + 2.(ch - 1)>](*cell*) = σ(*ch, f;* 1, 0, *cell*) (Req 4.16.12-23)

Cloudy pixels, forward.

[AST-MDS12-<60 + 2.(ch - 1)>](*cell*) = $\tilde{A}$ (*ch, f;* 1, 1, *cell*) (Req 4.16.12-24)

[AST-MDS12-<61 + 2.(ch - 1)>](*cell*) = σ(*ch, f;* 1, 1, *cell*) (Req 4.16.12-25)

[AST-MDS12-74](*cell*) = *PFF(f, land; cell)* [L2-INT-51] (Req 4.16.12-26)

[AST-MDS12-75](*cell*) = N0(*cell*) (Req 4.16.12-27)

[AST-MDS12-76](*cell*) = (ACLOUD) Percentage of pixels over <surface type>

(Req 4.16.12-28)

Averaged cloud parameters (ACLOUD), nadir view:

[AST-MDS12-77](*cell*) = *I_min(ir11, n, land;cell)* (Req 4.16.12-29)

[AST-MDS12-78](*cell*) = *I_min(ir12, n, land;cell)* (Req 4.16.12-30)

[AST-MDS12-79](*cell*) = *I_min(ir37, n, land;cell)* (Req 4.16.12-31)

[AST-MDS12-80](*cell*) = *I_min(v16, n, land;cell)* (Req 4.16.12-32)

[AST-MDS12-81](*cell*) = *I_min(v870, n, land;cell)* (Req 4.16.12-33)

[AST-MDS12-82](*cell*) = *I_min(v670, n, land;cell)* (Req 4.16.12-34)

[AST-MDS12-83](*cell*) = *I_min(v555, n, land;cell)* (Req 4.16.12-35)

Averaged cloud parameters (ACLOUD), forward view:

[AST-MDS12-84](*cell*) = *I_min(ir11, f, land;cell)* (Req 4.16.12-36)

[AST-MDS12-85](*cell*) = *I_min(ir12, f, land;cell)* (Req 4.16.12-37)

[AST-MDS12-86](*cell*) = *I_min(ir37, f, land;cell)* (Req 4.16.12-38)

[AST-MDS12-87](*cell*) = *I_min(v16, f, land;cell)* (Req 4.16.12-39)

[AST-MDS12-88](*cell*) = *I_min(v870, f, land;cell)* (Req 4.16.12-40)

[AST-MDS12-89](*cell*) = *I_min(v670, f, land;cell)* (Req 4.16.12-41)

[AST-MDS12-90](*cell*) = *I_min(v555, f, land;cell)* (Req 4.16.12-42)

Step 4.16.13 AST MDS#13: Sea Cell BT/TOA Record, 50 km cell

Record identified by cell number *l*, *m*.

[AST-MDS13-1](*l*, *m*) = [AST-MDS1-1](*l*, *m*) (Req 4.16.13-1)

[AST-MDS13-2](*l*, *m*) = -1 if (*N_sea(n; l, m)* = 0 and *N_sea(f; l, m)* = 0)
= 0 otherwise (Req 4.16.13-2)

[AST-MDS13-3](*l*, *m*) = (3 zero bytes) (Req 4.16.13-3)

[AST-MDS13-4](*l*, *m*) = *cell_lat(l, m)* [L2-INT-77] (Req 4.16.13-4)

[AST-MDS13-5](*l*, *m*) = *cell_long(l, m)* [L2-INT-78] (Req 4.16.13-5)

[AST-MDS13-6](*l*, *m*) = *across_track_mean*(0; *l, m)* [L2-INT-468] (Req 4.16.13-6)

[AST-MDS13-7](*l*, *m*) = *N_total(n; l, m)* [L2-INT-408] (Req 4.16.13-7)

[AST-MDS13-8](*l*, *m*) = *N_sea(n; l, m)* [L2-INT-407] (Req 4.16.13-8)

[AST-MDS13-9](*l*, *m*) = *pcs(n; l, m)* [L2-INT-409] (Req 4.16.13-9)

Clear pixels, nadir.

[AST-MDS13-<10 + 2.(ch - 1)>](*l*, *m*) = $\tilde{A}$ (ch, n; 0, 0, *l*, *m*) (Req 4.16.13-10)

[AST-MDS13-<11 + 2.(ch - 1)>](*l*, *m*) = σ(ch, n; 0, 0, *l*, *m*) (Req 4.16.13-11)

Cloudy pixels, nadir.

[AST-MDS13-<24 + 2.(ch - 1)>](*l*, *m*) = $\tilde{A}$ (ch, n; 0, 1, *l*, *m*) (Req 4.16.13-12)

[AST-MDS13-<25 + 2.(ch - 1)>](*l*, *m*) = σ(ch, n; 0, 1, *l*, *m*) (Req 4.16.13-13)

[AST-MDS13-38](*l*, *m*) = *PFF(n, sea; l, m)* [L2-INT-81] (Req 4.16.13-14)

Forward view:

[AST-MDS13-39](*l*, *m*) = *N_total(f; l, m)* [L2-INT-418] (Req 4.16.13-15)

[AST-MDS13-40](*l*, *m*) = *N_sea(f; l, m)* [L2-INT-417] (Req 4.16.13-16)

[AST-MDS13-41](*l*, *m*) = *pcs(f; l, m)* [L2-INT-419] (Req 4.16.13-17)

Clear pixels, forward.

[AST-MDS13-<42 + 2.(ch - 1)>](*l*, *m*) = $\tilde{A}$ (ch,f; 0, 0, *l*, *m*) (Req 4.16.13-18)

[AST-MDS13-<43 + 2.(ch - 1)>](*l*, *m*) = σ(ch, f; 0, 0, *l*, *m*) (Req 4.16.13-19)

Cloudy pixels, forward.

[AST-MDS13-<56 + 2.(ch - 1)>](*l*, *m*) = $\tilde{A}$ (ch, f; 0, 1, *l*, *m*) (Req 4.16.13-20)

[AST-MDS13-<57 + 2.(ch - 1)>](*l*, *m*) = σ(ch, f; 0, 1, *l*, *m*) (Req 4.16.13-21)

[AST-MDS13-70](*l*, *m*) = *PFF(f, sea; l, m)* [L2-INT-81] (Req 4.16.13-22)

[AST-MDS13-71](*l*, *m*) = N0(*l*, *m*) (Req 4.16.13-23)

[AST-MDS13-72](*l*, *m*) = (ACLOUD) Percentage of pixels over <surface type4

(Req 4.16.13-24)

Averaged cloud parameters (ACLOUD), nadir view:

[AST-MDS13-73](*l*, *m*) = *I_min(ir11, n, sea; l, m)* (Req 4.16.13-25)

[AST-MDS13-74](*l*, *m*) = *I_min(ir12, n, sea; l, m)* (Req 4.16.13-26)

[AST-MDS13-75](*l*, *m*) = *I_min(ir37, n, sea; l, m)* (Req 4.16.13-27)

[AST-MDS13-76](*l*, *m*) = *I_min(v16, n, sea; l, m)* (Req 4.16.13-28)

[AST-MDS13-77](*l*, *m*) = *I_min(v870, n, sea; l, m)* (Req 4.16.13-29)

[AST-MDS13-78](l, m) = I_min(v670, n, sea; l, m) (Req 4.16.13-30)

[AST-MDS13-79](l, m) = I_min(v555, n, sea; l, m) (Req 4.16.13-31)

Averaged cloud parameters (ACLOUD), forward view:

[AST-MDS13-80](l, m) = I_min(ir11, f, sea; l, m) (Req 4.16.13-32)

[AST-MDS13-81](l, m) = I_min(ir12, f, sea; l, m) (Req 4.16.13-33)

[AST-MDS13-82](l, m) = I_min(ir37, f, sea; l, m) (Req 4.16.13-34)

[AST-MDS13-83](l, m) = I_min(v16, f, sea; l, m) (Req 4.16.13-35)

[AST-MDS13-84](l, m) = I_min(v870, f, sea; l, m) (Req 4.16.13-36)

[AST-MDS13-85](l, m) = I_min(v670, f, sea; l, m) (Req 4.16.13-37)

[AST-MDS13-86](l, m) = I_min(v555, f, sea; l, m) (Req 4.16.13-38)

Step 4.16.14 AST MDS#14: Sea Cell BT/TOA Record, 17 km cell

Record identified by (k, l, m)

Clear pixels, nadir.

[AST-MDS14-1](k, l, m) = [AST-MDS2-1](k, l, m) (Req 4.16.14-1)

[AST-MDS14-2](k, l, m) = -1 if (N_sea(n; k, l, m) = 0 and N_sea(f; k, l, m) = 0)
= 0 otherwise (Req 4.16.14-2)

[AST-MDS14-3](k, l, m) = (3 zero bytes) (Req 4.16.14-3)

[AST-MDS14-4](k, l, m) = sub_cell_lat(k, l, m) [L2-INT-62] (Req 4.16.14-4)

[AST-MDS14-5](k, l, m) = sub_cell_long(k, l, m) [L2-INT-63] (Req 4.16.14-5)

[AST-MDS14-6](k, l, m) = across_track_mean(0; k, l, m) [L2-INT-459] (Req 4.16.14-6)

[AST-MDS14-7](k, l, m) = N_total(n; k, l, m) [L2-INT-403] (Req 4.16.14-7)

[AST-MDS14-8](k, l, m) = N_sea(n; k, l, m) [L2-INT-402] (Req 4.16.14-8)

[AST-MDS14-9](k, l, m) = pcs(n; k, l, m) [L2-INT-404] (Req 4.16.14-9)

[AST-MDS14-<10 + (ch - 1)>](k, l, m) = A(ch, n; 0, 0, k, l, m) (Req 4.16.14-10)

Cloudy pixels, nadir.

[AST-MDS14-<17 + (ch - 1)>](k, l, m) = A(ch, n; 0, 1, k, l, m) (Req 4.16.14-11)

[AST-MDS14-24](k, l, m) = PFF(n, sea; k, l, m) [L2-INT-82] (Req 4.16.14-12)

Forward view:

[AST-MDS14-25](k, l, m) = N_total(f; k, l, m) [L2-INT-413] (Req 4.16.14-13)

[AST-MDS14-26](k, l, m) = N_sea(f; k, l, m) [L2-INT-412] (Req 4.16.14-14)

[AST-MDS14-27](k, l, m) = pcs(f; k, l, m) [L2-INT-414] (Req 4.16.14-15)

Clear pixels, forward.

[AST-MDS14-<28 + (ch - 1)>](*k*, *l*, *m*) = A(ch, f; 0, 0, *k*, *l*, *m*) (Req 4.16.14-16)

Cloudy pixels, forward.

[AST-MDS14-<35 + (ch - 1)>](*k*, *l*, *m*) = A(ch, f; 0, 1, *k*, *l*, *m*) (Req 4.16.14-17)

[AST-MDS14-42](*k*, *l*, *m*) = *PFF(f, sea; k, l, m)* [L2-INT-82] (Req 4.16.14-18)

Step 4.16.15 AST MDS#15: Sea Cell BT/TOA Record, 10 arc minute cell

Record identified by (*k*, *cell)*

Clear pixels, nadir.

[AST-MDS15-1](*k*, *cell*) = [AST-MDS3-1](*k*, *cell*) (Req 4.16.15-1)

[AST-MDS15-2](*k*, *cell*) = -1 if (*N_sea(n; k, cell)* = 0 and *N_sea(f;k, cell)* = 0)
= 0 otherwise (Req 4.16.15-2)

[AST-MDS15-3](*k*, *cell*) = (3 zero bytes) (Req 4.16.15-3)

[AST-MDS15-4](*k*, *cell*) = *sub_cell_lat(k, cell)* [L2-INT-32] (Req 4.16.15-4)

[AST-MDS15-5](*k*, *cell*) = *sub_cell_long(k, cell)* [L2-INT-33] (Req 4.16.15-5)

[AST-MDS15-6](*k*, *cell*) = *across_track_mean*(0; *k*, *cell)* [L2-INT-359] (Req 4.16.15-6)

[AST-MDS15-7](*k*, *cell*) = *N_total(n; k, cell)* [L2-INT-303] (Req 4.16.15-7)

[AST-MDS15-8](*k*, *cell*) = *N_sea(n; k, cell)* [L2-INT-302] (Req 4.16.15-8)

[AST-MDS15-9](*k*, *cell*) = *pcs(n; k, cell)* [L2-INT-304] (Req 4.16.15-9)

[AST-MDS15-<10 + (ch - 1)>](*k*, *cell*) = *A(ch, n; 0, 0, k, cell)* (Req 4.16.15-10)

Cloudy pixels, nadir.

[AST-MDS15-<17 + (ch - 1)>](*k*, *cell*) = *A(ch, n; 0, 1, k, cell)* (Req 4.16.15-11)

[AST-MDS15-24](*k*, *cell*) = *PFF(n, sea; k, cell)* [L2-INT-52] (Req 4.16.15-12)

Forward view:

[AST-MDS15-25](*k*, *cell*) = *N_total(f; k, cell)* [L2-INT-313] (Req 4.16.15-13)

[AST-MDS15-26](*k*, *cell*) = *N_sea(f; k, cell)* [L2-INT-312] (Req 4.16.15-14)

[AST-MDS15-27](*k*, *cell*) = *pcs(f; k, cell)* [L2-INT-314] (Req 4.16.15-15)

Clear pixels, forward.

[AST-MDS15-<28 + (ch - 1)>](*k*, *cell*) = *A(ch, f; 0, 0, k, cell)* (Req 4.16.15-16)

Cloudy pixels, forward.

[AST-MDS15-<35 + (ch - 1)>](*k*, *cell*) = *A(ch, f; 0, 1, k, cell)* (Req 4.16.15-17)

[AST-MDS15-42](*k*, *cell*) = *PFF(f, sea; k, cell)* [L2-INT-52] (Req 4.16.15-18)

Step 4.16.16 AST MDS#16: Sea Cell BT/TOA Record, 30 arc minute cell

Record identified by cell number *cell*.

[AST-MDS16-1](*cell*) = [AST-MDS4-1](*cell*) (Req 4.16.16-1)

[AST-MDS16-2](*cell*) = -1 if (*N_sea(n; cell)* = 0 and *N_sea(f; cell)* = 0)
= 0 otherwise (Req 4.16.16-2)

[AST-MDS16-3](*cell*) = (3 zero bytes) (Req 4.16.16-3)

[AST-MDS16-4](*cell*) = *cell_lat(cell)* [L2-INT-47] (Req 4.16.16-4)

[AST-MDS16-5](*cell*) = *cell_long(cell)* [L2-INT-48] (Req 4.16.16-5)

[AST-MDS16-6](*cell*) = *across_track_mean(*0; *cell)* [L2-INT-368] (Req 4.16.16-6)

[AST-MDS16-7](*cell*) = *N_total(n; cell)* [L2-INT-308] (Req 4.16.16-7)

[AST-MDS16-8](*cell*) = *N_sea(n; cell)* [L2-INT-307] (Req 4.16.16-8)

[AST-MDS16-9](*cell*) = *pcs(n; cell)* [L2-INT-309] (Req 4.16.16-9)

Clear pixels, nadir.

[AST-MDS16-<10 + 2.(ch - 1)>](*cell*) = $\tilde{A}$ (*ch, n; 0, 0, cell*) (Req 4.16.16-10)

[AST-MDS16-<11 + 2.(ch - 1)>](*cell*) = σ(*ch, n; 0, 0, cell*) (Req 4.16.16-11)

Cloudy pixels, nadir.

[AST-MDS16-<24 + 2.(ch - 1)>](*cell*) = $\tilde{A}$ (ch, n; 0, 1, cell) (Req 4.16.16-12)

[AST-MDS16-<25 + 2.(ch - 1)>](*cell*) = σ(*ch, n; 0, 1, cell*) (Req 4.16.16-13)

[AST-MDS16-38](*cell*) = *PFF(n, sea; cell)* [L2-INT-5] (Req 4.16.16-14)

Forward view:

[AST-MDS16-39](*cell*) = *N_total(f; cell)* [L2-INT-318] (Req 4.16.16-15)

[AST-MDS16-40](*cell*) = *N_sea(f; cell)* [L2-INT-317] (Req 4.16.16-16)

[AST-MDS16-41](*cell*) = *pcs(f; cell)* [L2-INT-319] (Req 4.16.16-17)

Clear pixels, forward.

[AST-MDS16-<42 + 2.(ch - 1)>](*cell*) = $\tilde{A}$ (*ch,f; 0, 0, cell*) (Req 4.16.16-18)

[AST-MDS16-<43 + 2.(ch - 1)>](*cell*) = σ(*ch, f; 0, 0, cell*) (Req 4.16.16-19)

Cloudy pixels, forward.

[AST-MDS16-<56 + 2.(ch - 1)>](*cell*) = $\tilde{A}$ (*ch, f; 0, 1, cell*) (Req 4.16.16-20)

[AST-MDS16-<57 + 2.(ch - 1)>](*cell*) = σ(*ch, f; 0, 1, cell*) (Req 4.16.16-21)

[AST-MDS16-70](*cell*) = *PFF(f, sea; cell)* [L2-INT-51] (Req 4.16.16-22)

[AST-MDS16-71](*cell*) = N0(*cell*) (Req 4.16.16-23)

[AST-MDS16-72](*cell*) = (ACLOUD) Percentage of pixels over <surface type>
(Req 4.16.16-24)

Averaged cloud parameters (ACLOUD), nadir view:

[AST-MDS16-73](*cell*) = *I_min(ir11, n, sea; cell)* (Req 4.16.16-25)

[AST-MDS16-74](*cell*) = *I_min(ir12, n, sea; cell)* (Req 4.16.16-26)

[AST-MDS16-75](*cell*) = *I_min(ir37, n, sea; cell)* (Req 4.16.16-27)

[AST-MDS16-76](*cell*) = *I_min(v16, n, sea; cell)* (Req 4.16.16-28)

[AST-MDS16-77](*cell*) = *I_min(v870, n, sea; cell)* (Req 4.16.16-29)

[AST-MDS16-78](*cell*) = *I_min(v670, n, sea; cell)* (Req 4.16.16-30)

[AST-MDS16-79](*cell*) = *I_min(v555, n, sea; cell)* (Req 4.16.16-31)

Averaged cloud parameters (ACLOUD), forward view:

[AST-MDS16-80](*cell*) = *I_min(ir11, f, sea; cell)* (Req 4.16.16-32)

[AST-MDS16-81](*cell*) = *I_min(ir12, f, sea; cell)* (Req 4.16.16-33)

[AST-MDS16-82](*cell*) = *I_min(ir37, f, sea; cell)* (Req 4.16.16-34)

[AST-MDS16-83](*cell*) = *I_min(v16, f, sea; cell)* (Req 4.16.16-35)

[AST-MDS16-84](*cell*) = *I_min(v870, f, sea; cell)* (Req 4.16.16-36)

[AST-MDS16-85](*cell*) = *I_min(v670, f, sea; cell)* (Req 4.16.16-37)

[AST-MDS16-86](*cell*) = *I_min(v555, f, sea; cell)* (Req 4.16.16-38)

4.17 Module Definition: Output ECMWF Product

4.17.1 Functional Description

The ECMWF Averaged SST Product is written to the output medium. First the MPH, and SPH are written, then the Measurement data sets.

4.17.2 Interface Definition

See IODD Tables and Internal Parameter List

4.17.3 Detailed Structure

Step 4.17.1 MPH Record

As per PO-RS-MDA-GS-2009

Step 4.17.2 SPH Record

The SPH is identical to that for the AST product but with DSDs as per PO-RS-MDA-GS-2009

Step 4.17.3 MDS#1

The contents of each Meteo product record comprise the contents of the MDS3 record that corresponds to the same cell, together with the clear sea brightness temperatures from the corresponding MDS15 record, ordered to ensure that 4 byte quantities are aligned on 4-byte boundaries. It is assembled as follows.

Record identified by (k, cell)

[ECM-MDS1-1](*k*, *cell*) = [AST-MDS3-1](*k*, *cell*) (Req 4.17.1)

[ECM-MDS1-2](*k*, *cell*) = -1 if (*N_sea(n; k, cell)* = 0 and *N_sea(f; k, cell)* = 0)
= 0 otherwise (Req 4.17.2)

[ECM-MDS1-3](*k*, *cell*) = (3 zero bytes) (Req 4.17.3)

[ECM-MDS1-4](*k*, *cell*) = *sub_cell_lat(k, cell)* [L2-INT-32] (Req 4.17.4)

[ECM-MDS1-5](*k*, *cell*) = *sub_cell_long(k, cell)* [L2-INT-33] (Req 4.17.5)

[ECM-MDS1-12](*k*, *cell*) = *A(ir12, n; 0, 0, k, cell)* (Req 4.17.6)

[ECM-MDS1-13](*k*, *cell*) = *A(ir11, n; 0, 0, k, cell)* (Req 4.17.7)

[ECM-MDS1-14](*k*, *cell*) = *A(ir37, n; 0, 0, k, cell)* (Req 4.17.8)

[ECM-MDS1-15](*k*, *cell*) = *A(ir12, f; 0, 0, k, cell)* (Req 4.17.9)

[ECM-MDS1-16](*k*, *cell*) = *A(ir11, f; 0, 0, k, cell)* (Req 4.17.10)

[ECM-MDS1-17](*k*, *cell*) = *A(ir37, f; 0, 0, k, cell)* (Req 4.17.11)

[ECM-MDS1-6](*k*, *cell*) = *across_track_mean(*0; *k, cell)* [L2-INT-359] (Req 4.17.12)

[ECM-MDS1-7](*k*, *cell*) = *T_nadir(k, cell)* [L2-INT-54] (Req 4.17.13)

[ECM-MDS1-8](*k*, *cell*) = the smaller of *M(ir11, n; 0, 0, k, cell)*,
M(ir12, n; 0, 0, k, cell) (Req 4.17.14)

[ECM-MDS1-9](*k*, *cell*) = *T_dual(k, cell)* [L2-INT-56] (Req 4.17.15)

[ECM-MDS1-10](*k*, *cell*) = the smallest of *M(ir11, n; 0, 0, k, cell)*,
M(ir12, n; 0, 0, k, cell), *M(ir11, f; 0, 0, k, cell)*,
M(ir12, f; 0, 0, k, cell) (Req 4.17.16)

[ECM -MDS1-11](*k*, *cell*) = *ast_conf(0; k, cell)* (Req 4.17.17)

5 INTERNAL PARAMETER LIST

Parameter ID	Variable	Name	Type	Units	Field Size	Fields	Comments
L2-INT-1	φ_g(ig, j)	Tie point latitude	float	deg.	4	j = 0, 22	
L2-INT-2	λ_g(ig, j)	Tie point longitude	float	deg.	4	j = 0, 22	
L2-INT-3	β^n(ig,k)	tie scan solar elevation, nadir	float	deg.	4	k = 0, 10	
L2-INT-4	A^n(ig,k)	tie scan solar azimuth, nadir	float	deg.	4	k = 0, 10	
L2-INT-5	β^f(ig,k)	tie scan solar elevation, forward	float	deg.	4	k = 0, 10	
L2-INT-6	A^f(ig,k)	tie scan solar azimuth, forward	float	deg.	4	k = 0, 10	
L2-INT-7	y_g(ig)	tie scan y co-ordinate	sl	m	4		
L2-INT-8							
L2-INT-9							
L2-INT-10	y(i)	image scan y co-ordinate	sl	m	4		
L2-INT-11							
L2-INT-12							
L2-INT-13	γ^n(ig,k)	tie scan satellite elevation, nadir	float	deg.	4	k = 0, 10	
L2-INT-14	B^n(ig,k)	tie scan satellite azimuth, nadir	float	deg.	4	k = 0, 10	
L2-INT-15	γ^f(ig,k)	tie scan satellite elevation, forward	float	deg.	4	k = 0, 10	
L2-INT-16	B^f(ig,k)	tie scan satellite azimuth, forward	float	deg.	4	k = 0, 10	
L2-INT-17							
L2-INT-18	MJDT(4)	Scan UTC in MJD Format	4*sl	MJD	16		
L2-INT-19							
L2-INT-20	utc(l, m)	50 km cell UTC	double	days	8	per cell	
L2-INT-21	utc(k, l, m)	17 km sub-cell UTC	double	days	8	k = 0, 8	
L2-INT-22							
L2-INT-23	MJDP[0]/(1)	Scan UTC in processing format	double		8	1	
L2-INT-24	MJDP[1]/(2)	Scan UTC in processing format	double		8	1	
L2-INT-25	status	status flag	sl	n/a	4	1	
L2-INT-26	time(sg)	scan UTC	double	days	8	per sg	
L2-INT-27	scan(sg)	scan number corresponding to time(sg)	us	none	2	per sg	
L2-INT-28							
L2-INT-29							
L2-INT-30	utc(cell)	cell UTC	double	days	8	per cell	
L2-INT-31	utc(k, cell)	sub-cell UTC	double	days	8	k = 0, 8	
L2-INT-32	sub_cell_lat(k, cell)	sub-cell latitude	sl	μdeg	4	k = 0, 8	
L2-INT-33	sub_cell_long(k, cell)	sub-cell longitude	sl	μdeg	4	k = 0, 8	
L2-INT-34	sub_cell_band(k, cell)	sub-cell across-track band	ss	none	2	k = 0, 8	
L2-INT-36	S(ch, v; sf, cl, k, cell)	sub-cell total, ch = 1, ..., 7	sl	n/a	4		
L2-INT-37	M(ch, v; sf, cl, k, cell)	sub-cell pixel count, ch = 1, ..., 7	ss	none	2		
L2-INT-38	A(ch, v; sf, cl, k, cell)	sub-cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	sl	0.001K	4		
L2-INT-39	A(ch, v; sf, cl, k, cell)	sub-cell reflectance average (for visible channels ch = 4, 5, 6, 7)	ss	0.01%	2		
L2-INT-40	$\tilde{M}$ (ch, v; sf, cl, cell)	cell pixel count, ch = 1, ..., 7	ss	none	2		
L2-INT-41	$\tilde{A}$ (ch, v; sf, cl, cell)	cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	sl	0.001K	4		
L2-INT-42	$\tilde{A}$ (ch, v; sf, cl, cell)	cell reflectance average (for visible channels ch = 4, 5, 6, 7)	ss	0.01%	2		
L2-INT-43	σ(ch, v; sf, cl, cell)	standard deviation of the cell average	float	0.001K or 0.01%	4		
L2-INT-45	nadir_solar_el(k, cell)	nadir solar elevation for sub-cell)	float	degrees	4	k = 0, 8	
L2-INT-46	frwd_solar_el(k, cell)	frwd solar elevation for sub-cell	float	degrees	4	k = 0, 8	
L2-INT-47	cell_lat(cell)	cell latitude (30 arcmin)	sl	μdeg	4		
L2-INT-48	cell_long(cell)	cell longitude (30 arcmin)	sl	μdeg	4		
L2-INT-49	nadir_day(k, cell)	nadir view sub-cell day/night flag	ss	flag	2	k = 0, 8	
L2-INT-50	frwd_day(k, cell)	forward view sub-cell day/night flag	ss	flag	2	k = 0, 8	
L2-INT-51	PFF(v, sf; cell)	Pixel threshold failure flags word, 30 arc minute cell	ss	flags	2		

L2-INT-52	PFF(v, sf; k, cell)	Pixel threshold failure flags word, 10 arc minute sub-cell	ss	flags	2		
L2-INT-53	T_nadir(cell)	nadir view sst	ss	0.01K	2		
L2-INT-54	T_nadir(k, cell)	sub-cell nadir view sst	ss	0.01K	2	k = 0, 8	
L2-INT-55	T_dual(cell)	dual view sst	ss	0.01K	2		
L2-INT-56	T_dual(k, cell)	sub-cell dual view sst	ss	0.01K	2	k = 0, 8	
L2-INT-57	σ_nadir(ASST; cell)	standard deviation of nadir view ASST	ss	0.01K	2		
L2-INT-58	σ_dual(ASST; cell)	standard deviation of dual view ASST	ss	0.01K	2		
L2-INT-59		(Parameter deleted)		n/a	4		
L2-INT-60	band(j)	number of across track band (or strip)	sl	none	4	j = 0, 511	
L2-INT-61	map(j)	across-track mapping	sl	none	4	j = 0, 512	
L2-INT-62	sub_cell_lat(k, l, m)	sub-cell latitude (17 km)	sl	μdeg	4		
L2-INT-63	sub_cell_long(k, l, m)	sub-cell longitude	sl	μdeg	4		
L2-INT-64	sub_cell_band(k, l, m)	sub-cell across-track band	ss	none	2		
L2-INT-65							
L2-INT-66	S(ch, v; sf, cl, k, l, m)	17 km sub-cell total ch = 1, ..., 7	sl	n/a	4		
L2-INT-67	M(ch, v; sf, cl, k, l, m)	ch = 1, 2, 3, 4, 5, 6, 7	ss	none	2		
L2-INT-68	A(ch, v; sf, cl, k, l, m)	sub-cell brightness temperature average (For infra-red channels ch = 1, 2, 3)	sl	0.001K	4		
L2-INT-69	A(ch, v; sf, cl, k, l, m)	sub-cell reflectance average (For visible channels ch = 4, 5, 6, 7)	ss	0.01%	2		
L2-INT-70	$\tilde{M}$ (ch, v; sf, cl, l, m)	50 km cell pixel count, ch = 1, ..., 7	ss	none	2		
L2-INT-71	$\tilde{A}$ (ch, v; sf, cl, l, m)	50 km cell brightness temperature average (for infra-red channels ch = 1, 2, 3)	sl	0.001K	4		
L2-INT-72	$\tilde{A}$ (ch, v; sf, cl, l, m)	50 km cell reflectance average (for visible channels ch = 4, 5, 6, 7)	ss	0.01%	2		
L2-INT-73	σ(ch, v; sf, cl, l, m)	standard deviation of the 50 km cell average	float	0.001K or 0.01%	4		
L2-INT-74							
L2-INT-75	nadir_solar_el(k, l, m)	nadir solar elevation for sub-cell	float	degrees	4		
L2-INT-76	frwrd_solar_el(k, l, m)	frwrd solar elevation for sub-cell	float	degrees	4		
L2-INT-77	cell_lat(l, m)	cell latitude (50 km)	sl	μdeg	4		
L2-INT-78	cell_long(l, m)	cell longitude (50 km)	sl	μdeg	4		
L2-INT-79	nadir_day(k, l, m)	nadir view sub-cell day/night flag	ss	flag	2	k = 0, 8	
L2-INT-80	frwrd_day(k, l, m)	forward view sub-cell day/night flag	ss	flag	2	k = 0, 8	
L2-INT-81	PFF(v, sf; l, m)	Pixel threshold failure flags word, 50 km cell	ss	flags	2		
L2-INT-82	PFF(v, sf; k, l, m)	Pixel threshold failure flags word, 17 km sub-cell	ss	flags	2	k = 0, 8	
L2-INT-83	T_nadir(l, m)	nadir view sst	ss	0.01K	2		
L2-INT-84	T_nadir(k, l, m)	sub-cell nadir view sst	ss	0.01K	2	k = 0, 8	
L2-INT-85	T_dual(l, m)	dual view sst	ss	0.01K	2		
L2-INT-86	T_dual(k, l, m)	sub-cell dual view sst	ss	0.01K	2	k = 0, 8	
L2-INT-87	σ_nadir(ASST; l, m)	standard deviation of nadir view ASST	ss	0.01K	2		
L2-INT-88	σ_dual(ASST; l, m)	standard deviation of dual view ASST	ss	0.01K	2		
L2-INT-89		(Parameter deleted)					
L2-INT-90	NDVI(k, cell)	Averaged NDVI in 10-arcmin cells	ss	0.0001	2		
L2-INT-91	<NDVI>(cell)	mean NDVI	ss	0.0001	2		
L2-INT-92	σ(NDVI; cell)	standard deviation of NDVI	ss	0.0001	2		
L2-INT-93	N0(cell)	Number of pixels in NDVI average, half degree cell	us	none	2		
L2-INT-94	N1(k, cell)	Number of pixels in NDVI average, 10 arc min cells	us	none	2		
L2-INT-95	NDVI(k, l, m)	Averaged NDVI in 10-arcmin cells	ss	0.0001	2		
L2-INT-96	<NDVI>(l, m)	mean NDVI	ss	0.0001	2		
L2-INT-97	σ(NDVI; l, m)	standard deviation of NDVI	ss	0.0001	2		
L2-INT-98	N0(l, m)	Number of pixels in NDVI average, half degree cell	us	none	2		
L2-INT-99	N1(k, l, m)	Number of pixels in NDVI average, 10 arc min cells	us	none	2		

		Regridded data structures:					
L2-INT-100	nadir_fill_state(i, j)	nadir fill state indicator	byte	none	1	j = 0, 511	
L2-INT-101	I(ir12, n; i, j) I(1, n; i, j)	nadir ir12 Brightness Temperature, 12 µm nadir Brightness Temperature	ss	K/100	2	512	
L2-INT-102	I(ir11, n; i, j) I(2, n; i, j)	nadir ir11 Brightness Temperature 11 µm nadir Brightness Temperature	ss	K/100	2	512	
L2-INT-103	I(ir37, n; i, j) I(3, n; i, j)	nadir ir37 Brightness Temperature 3.7 µm nadir Brightness Temperature	ss	K/100	2	512	
L2-INT-104	I(v16, n; i, j) I(4, n; i, j)	nadir v16 Reflectance 1.6 µm nadir Reflectance	ss	%/100	2	512	
L2-INT-105	I(v870, n; i, j) I(5, n; i, j)	nadir v870 Reflectance 0.870 µm nadir Reflectance	ss	%/100	2	512	
L2-INT-106	I(v670, n; i, j) I(6, n; i, j)	nadir v670 Reflectance 0.670 µm nadir Reflectance	ss	%/100	2	512	
L2-INT-107	I(v555, n; i, j) I(7, n; i, j)	nadir v555 Reflectance 0.555 µm nadir Reflectance	ss	%/100	2	512	
L2-INT-110	frwd_fill_state(i, j)	frwd fill state indicator	byte	none	1	j = 0, 511	
L2-INT-111	I(ir12, f; i, j) I(1, f; i, j)	forward ir12 Brightness Temperature 12 µm forward Brightness Temperature	ss	K/100	2	512	
L2-INT-112	I(ir11, f; i, j) I(2, f; i, j)	forward ir11 Brightness Temperature 11 µm forward Brightness Temperature	ss	K/100	2	512	
L2-INT-113	I(ir37, f; i, j) I(3, f; i, j)	forward ir37 Brightness Temperature 3.7 µm forward Brightness Temperature	ss	K/100	2	512	
L2-INT-114	I(v16, f; i, j) I(4, f; i, j)	forward v16 Reflectance 1.6 µm forward Reflectance	ss	%/100	2	512	
L2-INT-115	I(v870, f; i, j) I(5, f; i, j)	forward v870 Reflectance 0.870 µm forward Reflectance	ss	%/100	2	512	
L2-INT-116	I(v670, f; i, j) I(6, f; i, j)	forward v670 Reflectance 0.670 µm forward Reflectance	ss	%/100	2	512	
L2-INT-117	I(v555, f; i, j) I(7, f; i, j)	forward v555 Reflectance 0.555 µm forward Reflectance	ss	%/100	2	512	
		regridded nadir information:					
L2-INT-120		nadir.band_edge_solar_elevation(i, k)	float	degrees	4	k = 0, 10	
L2-INT-121		nadir.band_edge_satellite_elevation(i, k)	float	degrees	4	k = 0, 10	
L2-INT-122		nadir.band_edge_solar_azimuth(i, k)	float	degrees	4	k = 0, 10	
L2-INT-123		nadir.band_edge_satellite_azimuth(i, k)	float	degrees	4	k = 0, 10	
L2-INT-124		nadir.band_centre_solar_elevation(i, k′)	float	degrees	4	k′ = 0, 9	
L2-INT-125		nadir.band_centre_satellite_elevation(i, k′)	float	degrees	4	k′ = 0, 9	
L2-INT-126		nadir.band_centre_solar_azimuth(i, k′)	float	degrees	4	k′ = 0, 9	
L2-INT-127		nadir.band_centre_satellite_azimuth(i, k′)	float	degrees	4	k′ = 0, 9	
L2-INT-128		nadir.band_centre_scan_times(i, k′)	double				
L2-INT-134	scn_nadir(ig, j)	nadir view instrument scan number	us	none	4	j = 0, 511	
L2-INT-135	pxl_nadir(ig, j)	nadir view instrument pixel number	us	none	4	j = 0, 511	
		regridded_frwd_info[.]					
L2-INT-140		frwd_band_edge_solar_elevation(i, k)	float	degrees	4	k = 0, 10	
L2-INT-141		frwd_band_edge_satellite_elevation(i, k)	float	degrees	4	k = 0, 10	
L2-INT-142		frwd.band_edge_solar_azimuth(i, k)	float	degrees	4	k = 0, 10	
L2-INT-143		frwd.band_edge_satellite_azimuth(i, k)	float	degrees	4	k = 0, 10	
L2-INT-144		frwd.band_centre_solar_elevation(i, k′)	float	degrees	4	k′ = 0, 9	
L2-INT-145		frwd.band_centre_satellite_elevation(i, k′)	float	degrees	4	k′ = 0, 9	
L2-INT-146		frwd.band_centre_solar_azimuth(i, k′)	float	degrees	4	k′ = 0, 9	
L2-INT-147		frwd.band_centre_satellite_azimuth(i, k′)	float	degrees	4	k′ = 0, 9	
L2-INT-148		frwd.band_centre_scan_times[10]	double				
L2-INT-149		.min_aux_temps[6]	float				
L2-INT-150		.max_aux_temps[6]	float				
L2-INT-151		.platform_mode	long int				
L2-INT-152		.pcd	long int				
L2-INT-153		.pixel_maps[2]	short int				
L2-INT-154	scn_frwrd(ig, j)	forward view instrument scan number	us	none	4	j = 0, 511	
L2-INT-155	pxl_frwrd(ig, j)	forward view instrument pixel number	us	none	4	j = 0, 511	

L2-INT-160	φ(i, j) image_latitude(i, j) image_lat(i, j)	image pixel latitude	float	deg.	4	j = 0, 511	
L2-INT-161	λ(i, j) image_longitude(i, j) image_long(I,j)	image pixel longitude	float	deg.	4	j = 0, 511	
L2-INT-171		GBTR confidence word, nadir view	ss	flags	2	j = 0, 511	
L2-INT-172		GBTR confidence word, forward view:	ss	flags	2	512	
L2-INT-173	gbtr_cloud_state_nadi r	GBTR cloud state flags, nadir view	ss	flags	2	512	
L2-INT-174	gbtr_cloud_state_frwr d	GBTR cloud state flags, forward view	ss	flags	2	512	
		Unpacked GBTR Confidence flags (nadir):					
L2-INT-200		nadir_blanking_pulse(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-201		nadir_cosmetic(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-202		nadir_scan_absent(i, j))	ss array	flag	2	j = 0, 511	
L2-INT-203		nadir_pixel_absent(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-204		nadir_packet_validation_error(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-205		nadir_zero_count(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-206		nadir_saturation(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-207		nadir_cal_out_of_range(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-208		nadir_calibration_unavailable(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-209		nadir_unfilled_pixel(i, j)	ss array	flag	2	j = 0, 511	
		Unpacked GBTR Confidence flags (forward):					
L2-INT-216		frwrd_blanking_pulse(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-217		frwrd_cosmetic(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-218		frwrd_scan_absent(i, j))	ss array	flag	2	j = 0, 511	
L2-INT-219		frwrd_pixel_absent(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-220		frwrd_packet_validation_error(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-221		frwrd_zero_count(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-222		frwrd_saturation(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-223		frwrd_cal_out_of_range(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-224		frwrd_calibration_unavailable(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-225		frwrd_unfilled_pixel(i, j)	ss array	flag	2	j = 0, 511	
		Unpacked GBTR cloud/land flags (nadir):					
L2-INT-232		nadir_land(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-233		nadir_cloud(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-234		nadir_sunglint(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-235		nadir_v16_histogram_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-236		nadir_v16_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-237		nadir_ir11_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-238		nadir_ir12_gross_cloud_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-239		nadir_ir11_ir12_thin_cirrus_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-240		nadir_ir37_ir12med_high_level_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-241		nadir_ir11_ir37_fog_low_stratus_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-242		nadir_ir11_ir12_view_diff_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-243		nadir_ir37_ir11_view_diff_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-244		nadir_ir11_ir12_histogram_test(i, j)	ss array	flag	2	j = 0, 511	
		Unpacked GBTR cloud/land flags (forward):					
L2-INT-248		frwrd_land(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-249		frwrd_cloud(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-250		frwrd_sunglint(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-251		frwrd_v16_histogram_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-252		frwrd_v16_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-253		frwrd_ir11_spatial_coherence_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-254		frwrd_ir12_gross_cloud_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-255		frwrd_ir11_ir12_thin_cirrus_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-256		frwrd_ir37_ir12med_high_level_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-257		frwrd_ir11_ir37_fog_low_stratus_test(i, j)	ss array	flag	2	j = 0, 511	

L2-INT-258		frwrd_ir11_ir12_view_diff_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-259		frwrd_ir37_ir11_view_diff_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-260		frwrd_ir11_ir12_histogram_test(i, j)	ss array	flag	2	j = 0, 511	
L2-INT-270		nadir_image_field(i, j)	ss	0.01 K	2	j = 0, 511	
L2-INT-271		combined_image_field(i, j)	ss	mixed	2	j = 0, 511	
L2-INT-272		gsst-confidence_word(i, j)	ss	flags	2	j = 0, 511	
		GSST confidence flags					
L2-INT-280		nadir_only_sst_valid(i, j) nadir_image_valid(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-281		nadir_only_sst_uses_ir37(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-282		dual_view_sst_valid(i, j) combined_image_valid(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-283		dual_view_sst_uses_ir37(i, j) combined_view_sst_uses_ir37(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-284		land(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-285		nadir_view_cloudy(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-286		nadir_view_blanking_pulse(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-287		nadir_view_cosmetic(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-288		frwrd_view_cloudy(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-289		frwrd_view_blanking_pulse(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-290		frwrd_view_cosmetic(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-291		gsst_v16_cloud_test(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-292		gsst_nadir_frwrd_cloud_test(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-293		gsst_ir11_histogtram_test(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-294		topographic_variance(i, j)	flag	n/a	2	j = 0, 511	
L2-INT-295		extended_land(i, j)	flag	n/a	2	j = 0, 511	
		Cell and sub-cell counts for half degree cells:					
L2-INT-301	N_land(n; k, cell)	total filled pixels over land for subcell	ss	none	2	k = 0, 8	
L2-INT-302	N_sea(n; k, cell)	total of filled pixels over sea for subcell	ss	none	2	k = 0, 8	
L2-INT-303	N_total(n; k, cell)	total of filled pixels for subcell, nadir view	ss	none	2	k = 0, 8	
L2-INT-304	pcs(n; k, cell)	percentage of cloudy pixels over sea	ss	0.01%	2	k = 0, 8	
L2-INT-305	pcl(n; k, cell)	percentage of cloudy pixels over land	ss	0.01%	2	k = 0, 8	
L2-INT-306	N_land(n; cell)	total filled pixels over land for cell	ss	none	2		
L2-INT-307	N_sea(n; cell)	total of filled pixels over sea for cell	ss	none	2		
L2-INT-308	N_total(n; cell)	total of filled pixels for cell, nadir view	ss	none	2		
L2-INT-309	pcs(n; cell)	percentage of cloudy pixels over sea	ss	0.01%	2		
L2-INT-310	pcl(n; cell)	percentage of cloudy pixels over land	ss	0.01%	2		
L2-INT-311	N_land(f; k, cell)	total filled pixels over land for sub-cell	ss	none	2	k = 0, 8	
L2-INT-312	N_sea(f; k, cell)	total of filled pixels over sea for subcell	ss	none	2	k = 0, 8	
L2-INT-313	N_total(f; k, cell)	total of filled pixels for subcell, frwrd view	ss	none	2	k = 0, 8	
L2-INT-314	pcs(f; k, cell)	percentage of cloudy pixels over sea	ss	0.01%	2	k = 0, 8	
L2-INT-315	pcl(f; k, cell)	percentage of cloudy pixels over land	ss	0.01%	2	k = 0, 8	
L2-INT-316	N_land(f; cell)	total filled pixels over land for cell	ss	none	2		
L2-INT-317	N_sea(f; cell)	total of filled pixels over sea for cell	ss	none	2		
L2-INT-318	N_total(f; cell)	total of filled pixels for cell, frwrd view	ss	none	2		
L2-INT-319	pcs(f; cell)	percentage of cloudy pixels over sea	ss	0.01%	2		
L2-INT-320	pcl(f; cell)	percentage of cloudy pixels over land	ss	0.01%	2		
L2-INT-321		(Parameter deleted)					
L2-INT-322		(Parameter deleted)					
L2-INT-323		(Parameter deleted)					
L2-INT-324		(Parameter deleted)					
L2-INT-325	across_track_band	across-track band	ss	none	2	per cell	
		land pixel histogram quantities:					
L2-INT-326		(Parameter deleted)					
L2-INT-327		(Parameter deleted)					
L2-INT-328	nadir_clear_land	total of clear land pixels, nadir view	sl	none	4	per cell	
L2-INT-329	frwd_clear_land	total of clear land pixels, forward view	sl	none	4	per cell	
L2-INT-330	nadir_cloudy_land	total of cloudy land pixels, nadir view	sl	none	4	per cell	

L2-INT-331	frwrd_cloudy_land	total of cloudy land pixels, forward view	sl	none	4	per cell	
L2-INT-332	nadir_hist_land	nadir histogram (land cell)	ss	none	2	1000	
L2-INT-333	frwrd_hist_land	forward histogram (land cell)	ss	none	2	1000	
L2-INT-334		(Parameter deleted)					
		nadir output quantities:					
L2-INT-335	bt_cloud_top	cloud top temperatuere (over land)	ss	0.01K	2	per cell	
L2-INT-336	bt_percent_cloudy	percentage cloudy pixels (over land)	ss	0.01%	2	per cell	
		forward output quantities:					
L2-INT-337	bt_cloud_top	cloud top temperatuere (over land)	ss	0.01K	2	per cell	
L2-INT-338	bt_percent_cloudy	percentage cloudy pixels (over land)	ss	0.01%	2	per cell	
		sea pixel histogram quantities:					
L2-INT-342		(Parameter deleted)					
L2-INT-343		(Parameter deleted)					
L2-INT-344	nadir_clear_sea	total of clear sea pixels, nadir view	sl	none	4	per cell	
L2-INT-345	frwrd_clear_sea	total of clear sea pixels, forward view	sl	none	4	per cell	
L2-INT-346	nadir_cloudy_sea	total of cloudy sea pixels, nadir view	sl	none	4	per cell	
L2-INT-347	frwrd_cloudy_sea	total of cloudy sea pixels, forward view	sl	none	4	per cell	
L2-INT-348	nadir_hist_sea	nadir histogram (sea cell)	ss	none	2	1000	
L2-INT-349	frwrd_hist_sea	forward histogram (sea cell)	ss	none	2	1000	
L2-INT-350		(Parameter deleted)					
		nadir output quantities:					
L2-INT-351	bt_cloud_top	cloud top temperatuere (over sea)	ss	0.01K	2	per cell	
L2-INT-352	bt_percent_cloudy	percentage cloudy pixels (over sea)	ss	0.01%	2	per cell	
		frwd output quantities:					
L2-INT-353	bt_cloud_top	cloud top temperatuere (over sea)	ss	0.01K	2	per cell	
L2-INT-354	bt_percent_cloudy	percentage cloudy pixels (over sea)	ss	0.01%	2	per cell	
L2-INT-355	N(v; sf, cl, k, cell)	sub-cell filled pixel count	ss	none	2		
L2-INT-356	band_sum(k, cell)	cumulative across-track band sum	sl	none	4		
L2-INT-357	mean_band(k, cell)	mean across-track band number	ss	none	2		
L2-INT-358	across_track_sum(sf; k, cell)	cumulative sum of across-track pixel index	sl	none	4	k = 0, 8	
L2-INT-359	across_track_mean(sf; k, cell)	mean across-track pixel index, subcell k	ss	none	2	k = 0, 8	
L2-INT-360	a(i, j, q)	averaged sst retrieval a coefficients	float	mixed	4	90	
L2-INT-361	b(i, j, q)	averaged sst retrieval b coefficients	float	mixed	4	120	
L2-INT-362	c(i, j, q)	averaged sst retrieval c coefficients	float	mixed	4	150	
L2-INT-363	d(i, j, q)	averaged sst retrieval d coefficients	float	mixed	4	210	
L2-INT-364	ast_conf(sf; k, cell)	AST confidence word for sub-cell	sl	flags	4	k = 0, 8	
L2-INT-365	ast_conf(sf; cell)	AST confidence word for cell	sl	flags	4	per cell	
L2-INT-366		nadir_asst_uses_ir37(k, cell)	ss	flag	2	k = 0, 8	
L2-INT-367		dual_asst_uses_ir37(k, cell)	ss	flag	2	k = 0, 8	
L2-INT-368	across_track_mean(sf; cell)	mean across-track pixel index, cell	ss	none	2	per cell	
L2-INT-369	sst_mean_pixel(sf; cell)	mean across-track pixel index, cell	ss	none	2	per cell	
		Cell and sub-cell counts for 50 km cells:					
L2-INT-401	N_land(n; k, l, m)	total filled pixels over land for subcell	ss	none	2	k = 0, 8	
L2-INT-402	N_sea(n; k, l, m)	total of filled pixels over sea for subcell	ss	none	2	k = 0, 8	
L2-INT-403	N_total(n; k, l, m)	total of filled pixels for subcell, nadir view	ss	none	2	k = 0, 8	
L2-INT-404	pcs(n; k, l, m)	percentage of cloudy pixels over sea	ss	0.01%	2	k = 0, 8	
L2-INT-405	pcl(n; k, l, m)	percentage of cloudy pixels over land	ss	0.01%	2	k = 0, 8	
L2-INT-406	N_land(n; l, m)	total filled pixels over land for cell	ss	none	2		
L2-INT-407	N_sea(n; l, m)	total of filled pixels over sea for cell	ss	none	2		
L2-INT-408	N_total(n; l, m)	total of filled pixels for cell, nadir view	ss	none	2		
L2-INT-409	pcs(n; l, m)	percentage of cloudy pixels over sea	ss	0.01%	2		
L2-INT-410	pcl(n; l, m)	percentage of cloudy pixels over land	ss	0.01%	2		
L2-INT-411	N_land(f; k, l, m)	total filled pixels over land for sub-cell	ss	none	2	k = 0, 8	
L2-INT-412	N_sea(f; k, l, m)	total of filled pixels over sea for subcell	ss	none	2	k = 0, 8	
L2-INT-413	N_total(f; k, l, m)	total of filled pixels for subcell, frwrd view	ss	none	2	k = 0, 8	
L2-INT-414	pcs(f; k, l, m)	percentage of cloudy pixels over sea	ss	0.01%	2	k = 0, 8	

L2-INT-415	pcl(f; k, l, m)	percentage of cloudy pixels over land	ss	0.01%	2	k = 0, 8	
L2-INT-416	N_land(f; l, m)	total filled pixels over land for cell	ss	none	2		
L2-INT-417	N_sea(f; l, m)	total of filled pixels over sea for cell	ss	none	2		
L2-INT-418	N_total(f; l, m)	total of filled pixels for cell, frwrd view	ss	none	2		
L2-INT-419	pcs(f; l, m)	percentage of cloudy pixels over sea	ss	0.01%	2		
L2-INT-420	pcl(f; l, m)	percentage of cloudy pixels over land	ss	0.01%	2		
L2-INT-421		(Parameter deleted)					
L2-INT-422		(Parameter deleted)					
L2-INT-423		(Parameter deleted)					
L2-INT-424		(Parameter deleted)					
L2-INT-425	across_track_band	across-track band	ss	none	2	per cell	
		land pixel histogram quantities:					
L2-INT-426		(Parameter deleted)					
L2-INT-427		(Parameter deleted)					
L2-INT-428	nadir_clear_land	total of clear land pixels, nadir view	sl	none	4	per cell	
L2-INT-429	frwrd_clear_land	total of clear land pixels, forward view	sl	none	4	per cell	
L2-INT-430	nadir_cloudy_land	total of cloudy land pixels, nadir view	sl	none	4	per cell	
L2-INT-431	frwrd_cloudy_land	total of cloudy land pixels, forward view	sl	none	4	per cell	
L2-INT-432	nadir_hist_land	nadir histogram (land cell)	ss	none	2	1000	
L2-INT-433	frwrd_hist_land	forward histogram (land cell)	ss	none	2	1000	
L2-INT-434		(Parameter deleted)					
		nadir output quantities:					
L2-INT-435	bt_cloud_top	cloud top temperatuere (over land)	ss	0.01K	2	per cell	
L2-INT-436	bt_percent_cloudy	percentage cloudy pixels (over land)	ss	0.01%	2	per cell	
		frwd output quantities:					
L2-INT-437	bt_cloud_top	cloud top temperatuere (over land)	ss	0.01K	2	per cell	
L2-INT-438	bt_percent_cloudy	percentage cloudy pixels (over land)	ss	0.01%	2	per cell	
		sea pixel histogram quantities:					
L2-INT-442		(Parameter deleted)					
L2-INT-443		(Parameter deleted)					
L2-INT-444	nadir_clear_sea	total of clear sea pixels, nadir view	sl	none	4	per cell	
L2-INT-445	frwrd_clear_sea	total of clear sea pixels, forward view	sl	none	4	per cell	
L2-INT-446	nadir_cloudy_sea	total of cloudy sea pixels, nadir view	sl	none	4	per cell	
L2-INT-447	frwrd_cloudy_sea	total of cloudy sea pixels, forward view	sl	none	4	per cell	
L2-INT-448	nadir_hist_sea	nadir histogram (sea cell)	ss	none	2	1000	
L2-INT-449	frwrd_hist_sea	forward histogram (sea cell)	ss	none	2	1000	
L2-INT-450		(Parameter deleted)					
		nadir output quantities:					
L2-INT-451	bt_cloud_top	cloud top temperatuere (over sea)	ss	0.01K	2	per cell	
L2-INT-452	bt_percent_cloudy	percentage cloudy pixels (over sea)	ss	0.01%	2	per cell	
		frwd output quantities:					
L2-INT-453	bt_cloud_top	cloud top temperature (over sea)	ss	0.01K	2	per cell	
L2-INT-454	bt_percent_cloudy	percentage cloudy pixels (over sea)	ss	0.01%	2	per cell	
L2-INT-455	N(v; sf, cl, k, l, m)	sub-cell filled pixel count	ss	none	2		
L2-INT-456	band_sum(k, l, m)	cumulative across-track band sum	sl	none	4		
L2-INT-457	mean_band(k, l, m)	mean across-track band number	ss	none	2		
L2-INT-458	across_track_sum(sf; k, l, m)	cumulative sum of across-track pixel index	sl	none	4	k = 0, 8	
L2-INT-459	across_track_mean(sf; k, l, m)	mean across-track pixel index, subcell k	ss	none	2	k = 0, 8	
L2-INT-460	a(i, j, q)	averaged sst retrieval a coefficients	float	mixed	4		
L2-INT-461	b(i, j, q)	averaged sst retrieval b coefficients	float	mixed	4		
L2-INT-462	c(i, j, q)	averaged sst retrieval c coefficients	float	mixed	4		
L2-INT-463	d(i, j, q)	averaged sst retrieval d coefficients	float	mixed	4		
L2-INT-464	ast_conf(sf; k, l, m)	AST confidence word for sub-cell	sl	flags	4	k = 0, 8	
L2-INT-465	ast_conf(sf; l, m)	AST confidence word for cell l, m	sl	flags	4	per cell	
L2-INT-466		nadir_asst_uses_ir37(k, l, m)	ss	flag	2	k = 0, 8	
L2-INT-467		dual_asst_uses_ir37(k, l, m)	ss	flag	2	k = 0, 8	
L2-INT-468	across_track_mean(sf; l, m)	mean across-track pixel index, cell l, m	ss	none	2	per cell	

L2-INT-469	sst_mean_pixel(sf; l, m)	mean across-track pixel index, cell l, m	ss	none	2	per cell	
L2-INT-470		vegetation_class(lat_index, lon_index)	ss	none	2	360 × 720	
L2-INT-471		vegetation_fraction(lat_index, lon_index)	ss	0.001	4	360 × 720	
L2-INT-472		precipitable_water(lat_index, lon_index)	ss	0.01 mm	2	360 × 720	
L2-INT-473		Topographic_flag(lat_index, lon_index)	ss	none	2	360 × 720	
L2-INT-474	lat_index	Latitude index: lat_index = 0, …359	sl	none	4	1	
L2-INT-475	lon_index	Longitude index: lon_index = 0, 719	sl	none	4	1	
L2-INT-489	disp_lat_index	Displaced latitude index: = 0, …359	sl	none	4	1	
L2-INT-490	disp_lon_index	Displaced longitude index: = 0, 719	sl	none	4	1	
L2-INT-476	month	Index to month: month = 0, 11	sl	none	4	1	
L2-INT-477	sun_elev	Solar elevation at land pixel	float	degrees	4	1	
L2-INT-478	sat_elev	Satellite elevation at land pixel	float	degrees	4	1	
L2-INT-479	night	Day/night flag	sl	none	2	1	
L2-INT-480	n	Non-linear exponent	float	none	4	1	
L2-INT-481	f	Vegetation fraction at pixel	float	none	4	1	
L2-INT-482	pw	Precipitable water at pixel	float	cm	4	1	
L2-INT-482	coeff(class, i, j, 0)	Sub-array of coefficients A0	float	0.01K	4	64	
L2-INT-484	coeff(class, i, j, 1)	Sub-array of coefficients A1	float	none	4	64	
L2-INT-485	coeff(class, i, j. 2)	Sub-array of coefficients A2	float	none	4	64	
L2-INT-486	w	Interpolation weight	float	none	4	1	
L2-INT-487	a(k)	Retrieval coefficients for pixel	float	mixed	4	1	
L2-INT-488	lst	Land surface temperature at pixel	float	0.01K	4	1	
		Variables for LST in 30 arc minute cells:					
L2-INT-491	sub_cell_index(k, cell)	Along-track index representative of sub-cell	sl	none	4	k = 0, 8	
L2-INT-492	T_land(k, cell)	Land surface temperature in sub-cell k.	ss	0.01 K	4	k = 0, 8	
L2-INT-493	T_land(cell)	Averaged land surface temperature in cell.	ss	0.01 K	4	per cell	
L2-INT-494	σ_land(cell)	Standard deviation of Averaged LST	ss	0.01 K	4	per cell	
		Variables for LST in 50 km cells:					
L2-INT-495	sub_cell_index(k, l, m)	Along-track index representative of sub-cell	sl	none	4	k = 0, 8	
L2-INT-496	T_land(k, l, m)	Land surface temperature in sub-cell k.	ss	0.01 K	4	k = 0, 8	
L2-INT-497	T_land(l, m)	Averaged land surface temperature in cell.	ss	0.01 K	4	per cell	
L2-INT-498	σ_land(l, m)	Standard deviation of Averaged LST	ss	0.01 K	4	per cell	

Table 5-1: Internal Parameter summary list