REVIEW

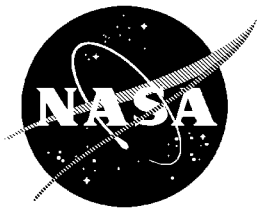
430-15-01-002-3
CSC 10040794

APPLIED ENGINEERING AND TECHNOLOGY DIRECTORATE

Landsat 7 System Calibration Parameter File Definition

Revision 3

April 1999



National Aeronautics and Space Administration

Goddard Space Flight Center
Greenbelt, Maryland

REVIEW

Landat 7 System Calibration Parameter File Definition

Revision 3

April1999

Prepared by:

R. Irish Date
Senior Systems Engineer
Landsat 7 Project Science Office
Science Systems and Applications, Inc.

B. Markam Date
Lead Calibration Scientist
Landsat 7 Processing System
GSFC, Code 923

Concurred by:

K. Michael Date
External Interface Manager
ESDIS Project
GSFC, Code 423

R. Schweiss Date
Deputy GS Implementation Manager
GSFC, Code 586

D. Williams Date
Project Scientist
Landsat 7 Project
GSFC, Code 923.0

Goddard Space Flight Center
Greenbelt, Maryland

Preface

This document is controlled by the Landsat 7 Ground System Configuration Control Board (CCB) and may be updated by a Document Change Notice (DCN) or a revision. Comments and questions regarding this document should be directed to:

Landsat 7 Image Assessment System Project
Code 586
Goddard Space Flight Center
Greenbelt, MD 20771

Abstract

This document describes the contents of the calibration parameter file (CPF) generated by the Image Assessment System (IAS), an element of the Landsat 7 Ground Segment. The IAS periodically performs radiometric and geometric calibration and updates the CPF. This file is stamped with applicability dates and sent to the Earth Resources Observation System (EROS) Data Center (EDC) Distributed Active Archive Center (DAAC) for storage and eventual bundling with outbound Level 0 reformatted (0R) products. The CPF also is sent to international ground stations via the Landsat 7 Mission Operations Center. The CPF supplies the radiometric and geometric correction parameters required during Level 1 processing to create superior products of uniform consistency across the Landsat 7 system.

Keywords: *calibration parameter file (CPF), Image Assessment System (IAS), EROS Data Center Distributed Active Archive Center (EDC DAAC)*

Change Information Page

List of Effective Pages	
Page Number	**Issue**
Title page	Revision 3
iii through ix	Revision 3
1-1 through 1-4	Revision3
2-1 through 2-123	Revision 3
3-1 through 3-26	Revision 3
A-1 through A-2	Revision 3

Document History			
Document Number	**Status/Issue**	**Publication Date**	**CCR Number**
430-15-01-002-0	Signature	September 1997	Signature
430-15-01-002-0 Revision 1	Signature	February 1998	Signature
430-15-01-002-2 Revision 2	Signature	July 1998	IAS980070 IAS980071 IAS980078 IAS980080 IAS980098
430-15-01-002-3 Revision 3	Review	April 1999	GS CCR 60 GS CCR XX

Table of Contents

Section 1. Introduction

Section 2. CPF Parameters

Section 3. CPF ODL

Acronyms and Abbreviations

Section 1. Introduction

This document describes the contents of the calibration parameter file (CPF) generated by the Image Assessment System (IAS), an element of the Landsat 7 Ground Segment. The IAS is responsible for offline assessment of image quality to ensure compliance with the radiometric and geometric requirements of the spacecraft and the Enhanced Thematic Mapper Plus (ETM+) sensor throughout the mission's life.

In addition to its assessment functions, the IAS is responsible for the radiometric and geometric calibration of the Landsat 7 satellite and ETM+. The IAS periodically performs radiometric and geometric calibration and updates the CPF. This file is stamped with applicability dates and sent to the Earth Resources Observation System (EROS) Data Center (EDC) Distributed Active Archive Center (DAAC) for storage and eventual bundling with outbound Level 0 reformatted (0R) products. The CPF also is sent to international ground stations via the Landsat 7 Mission Operations Center. The CPF supplies the radiometric and geometric correction parameters required during Level 1 processing to create superior products of uniform consistency across the Landsat 7 system.

1.1 Document Organization

Section 1 provides an introduction to the CPF. It describes the CPF structure and language; the CPF updates, time stamps, and file-naming conventions; and the attributes used to characterize the calibration parameters. Section 2 contains a table that lists and describes the actual CPF parameters. Section 3 presents the syntax of the CPF Object Description Language (ODL) and a CPF construct that illustrates the actual appearance of the file. The to be submitted (TBS) items in Sections 2 and 3 indicate where actual values for the parameters are not yet known. This document will not be updated to remove the TBS items, as the actual prelaunch and postlaunch CPFs will contain the most recent and accurate data available for these parameters. Following Section 3 is a list of the acronyms and abbreviations used in this document.

1.2 Applicable Document

A useful ODL document is the Jet Propulsion Laboratory (JPL), California Institute of Technology's *Planetary Data System Standards Reference, Version 3.2*, Chapter 12, Object Description Language Specification and Usage, July 24, 1995.

This document can be accessed online at URL http://pds.jpl.nasa.gov/stdref/chap12.htm.

1.3 File Structure

All parameters are stored as American Standard Code for Information Interchange (ASCII) text using the ODL syntax developed by JPL. ODL is a tagged keyword language developed to provide a human-readable data structure to encode data for simplified interchange. The ODL interpreter developed by JPL may, in certain cases, provide for the handling of lexical elements (for example, building blocks) that are included in the Consultative Committee for Space Data

Systems (CCSDS) specification of the Parameter Value Language (PVL). PVL is a superset of ODL. The IAS CPF is a pure ODL implementation without any PVL extensions.

The body of the file is composed of two statement types:

1. Attribute assignment statement used to assign values to parameters
2. Group statements used to aid in file organization and enhance parsing granularity of parameter sets

ODL details can be found in the *Planetary Data System Standards Reference*.

1.4 Calibration Parameter File Updates

The IAS will update and distribute the CPF at least every 90 days. Updates will likely be more frequent during early orbit checkout and also will occur between the regular 90-day cycles, as necessary. Irregular updates, however, will not affect the 90-day schedule. The timed release of a new CPF must be maintained because of the Universal Time Code corrected (UT1) time corrections and pole wander predictions included in the file. These parameters span a 180-day interval time centered on the effective start date of the new IAS CPF.

1.4.1 Time Stamps

The CPF is time-stamped with an effective date range. The first two parameters in the file—Effective_Date_Begin and Effective_Date_End—designate the range and are in YYYY-MM-DD format. The Effective_Date_End for the most recent parameter file is its Effective_Date_Begin plus 90 days. After this date, the file is without applicable UT1 time predictions. The Earth Observing System Data and Information System (EOSDIS) Core System (ECS) maintains a database of CPF names and their effective dates for associating product orders with the appropriate parameter files. The parameter file that accompanies an order has an effective date range that includes the acquisition date of the image ordered.

1.4.2 File Naming Conventions

Through the course of the mission, a serial collection of CPFs will be generated and sent to the EDC DAAC for coupling to 0R distribution products. The probability exists that a CPF will be replaced due to improved calibration parameters for a given period or perhaps file error. The need for unique file sequence numbers becomes necessary as file contents change. The following file-naming procedure is used by the IAS to name the CPF.

L7CPFyyyymmdd_yyyymmdd.nn

where L7 = Constant for Landsat 7

CPF = 3-letter CPF designator

yyyy = 4-digit effectivity starting year

mm = 2-letter effectivity starting month

dd = 2-letter effectivity starting day

_	=	Effectivity starting/ending date separator
yyyy	=	4-digit effectivity ending year
mm	=	2-letter effectivity ending month
dd	=	2-letter effectivity ending day
nn	=	Sequence number for this file

As an example, suppose four calibration files were created by the IAS on 90-day intervals and sent to the EDC DAAC during the first year of the mission. Further suppose that the first file was updated twice and the second and third files were updated once. The assigned file names would be as follows:

File 1 L7CPF19980601_19980829.00
L7CPF19980601_19980829.01
L7CPF19980601_19980829.02

File 2 L7CPF19980830_19981127.01
L7CPF19980830_19981127.02

File 3 L7CPF19981128_19990225.01
L7CPF19981128_19990225.02

File 4 L7CPF19990226_19990526.01

It is worth noting the **00** sequence number assigned to the original CPF. This reserve sequence number uniquely identifies the prelaunch CPF. Sequence numbers for subsequent time periods all begin with 01. New versions or updates are incremented by one.

This example assumes the effectivity dates do not change. The effectivity date range for a file can change, however, if a specific problem (e.g., detector outage) is discovered somewhere within the nominal 90-day effectivity range. Assuming this scenario, two CPFs with new names and effectivity date ranges are spawned for the time period under consideration. The **effective_date_end** for a new pre-problem CPF would change to the day before the problem occurred. The **effective_date_begin** remains unchanged. A post-problem CPF with a new file name would be created with an **effective_date_begin** corresponding to the imaging date the problem occurred. The **effective_date_end** assigned would be the original **effective_date_end** for the time period under consideration. New versions of all other CPFs affected by the erroneous parameter also would be created.

Using this example, suppose a dead detector is discovered to have occurred on January 31, 1999. Two new CPFs are created that supersede the time period represented by file number three, version 2, and a new version of file number four is created. The new file names and sequence numbers become

File 3 L7CPF19981128_19990225.01
L7CPF19981128_19990225.02

L7CPF19981128_19990131.03
L7CPF19990201_19990225.03

File 4 L7CPF19990226_19990526.01
L7CPF19990226_19990526.02

1.5 File Content Description

Each parameter entry is characterized by five attributes:

1. Parameter group—Identifies a related set of parameters.
2. Parameter name—Uniquely identifies and describes the content of each parameter.
3. Value type—Describes the parameter as either static or dynamic. A static value remains unchanged over the mission's life. A dynamic value will change or has the potential to change over the life of the mission. Significant changes to dynamic values trigger a CPF update.
4. Data type—Referred to using Hierarchical Data Format (HDF) number type nomenclature, type#, where type is either char (character), int (integer), or float (floating point), and # is a decimal count of the number of bits used to represent the data type. The type mnemonics int and char may be preceded by the letter u, indicating an unsigned value. For example, the data type uint32 refers to an unsigned 32-bit integer value. Data types relevant to the CPF are as follows:

Data Type	HDF Nomenclature
8-bit character	char8
8-bit unsigned integer	uint8
16-bit signed integer	int16
32-bit signed integer	int32
32-bit floating point number	float32
64-bit floating point number	float64

5. Description—Short description of the parameter, its format, and its nominal or expected value(s).

Section 2. CPF Parameters

The following table lists the CPF parameters.

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
FILE_ATTRIBUTES	Effective_Date_Begin	Dynamic	char8	Effective start date for this file Valid format: YYYY-MM-DD, where YYYY = 1998-2050, MM = 01-12, and DD = 01-31	Yes	Yes	IAS
FILE_ATTRIBUTES	Effective_Date_End	Dynamic	char8	Effective end date for this file Valid format: YYYY-MM-DD, where YYYY = 1998-2050, MM = 01-12, and DD = 01-31	Yes	Yes	IAS
FILE_ATTRIBUTES	CPF_File_Name	Dynamic	char8	Original file name assigned by IAS Valid format: L7CPFyyyymmdd-yyyymmdd.nn where yyyymmdd = effective start date and effective end date, respectfully, and nn = incrementing version for within a quarter (01-99)	Yes	Yes	IAS
EARTH_CONSTANTS	Ellipsoid_Name	Static	char8	Name of ellipsoid used to represent semi-major and semi-minor axes of Earth Valid format: SSSSS, where SSSSS = WGS84	Yes	Yes	EDC
EARTH_CONSTANTS	Semi_Major_Axis	Static	float64	Earth semi-major axis; distance in meters from center of Earth to equator Valid format: NNNNNNN.NNN, where NNNNNNN.NNN = 6378137.000	Yes	Yes	EDC
EARTH_CONSTANTS	Semi_Minor_Axis	Static	float64	Earth semi-minor axis; distance in meters from center of Earth to poles Valid format: NNNNNNN.NNN, where NNNNNNN.NNN = 6356752.314	Yes	Yes	EDC
EARTH_CONSTANTS	Ellipticity	Static	float64	Ratio describing polar flattening or Earth's deviation from an exact sphere (WGS84 standard) Valid format: N.NNNNNNNNN, where N.NNNNNNNNN = (1/298.257223563)	Yes	Yes	EDC

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
EARTH_CONSTANTS	Eccentricity	Static	float64	Number describing the Earth ellipsoid eccentricity squared (WGS84 standard) Valid format: N.NNNNNNNNNNNNN, where N.NNNNNNNNNNNNN = 0.00669437999013	Yes	Yes	EDC
EARTH_CONSTANTS	Earth_Spin_Rate	Static	float64	Earth's diurnal spin rate in radians per second Valid format: NN.NNNNNNNNENNN, where NN.NNNNNNNNENNN = 72.92115855E-06	Yes	Yes	EDC
EARTH_CONSTANTS	Gravity_Constant	Static	float64	Universal gravitational constant times mass of Earth. This parameter is given in units of meters cubed per second squared (m^3/s^2). Valid format: N.NNNNNNENN, where N.NNNNNNENN = 3.986005E14	Yes	Yes	EDC
EARTH_CONSTANTS	J2_Earth_Model_ Term	Static	float64	Term that describes Earth's spherical harmonic Valid format: NNNN.NNESNN, where NNNN.NNESNN = 1082.64E-06	Yes	Yes	EDC
ORBIT_PARAMETERS	WRS_Cycle_Days	Static	uint8	Time period, in days, required for satellite to view Earth once Valid format: NN, where NN = 16	No	Yes	IAS
ORBIT_PARAMETERS	WRS_Cycle_Orbits	Static	uint8	Number of orbits or paths in a complete World Reference System (WRS) cycle Valid format: NNN, where NNN = 233	No	Yes	IAS
ORBIT_PARAMETERS	Scenes_Per_Orbit	Static	uint8	Number of scenes or row locations per orbit Valid format: NNN, where NNN = 248	No	Yes	IAS
ORBIT_PARAMETERS	Orbital_Period	Static	float64	Time required, in seconds, to complete one orbit Valid format: NNNN.NNNN, where NNNN.NNNN = 5933.0472	No	Yes	LMC System Spec
ORBIT_PARAMETERS	Angular_Momentum	Static	float64	Angular momentum in orbit, specified in meters squared per second Valid format: NN.NNNNNNEN, where NN.NNNNNNEN = 53.104278E9	No	Yes	LMC System Spec

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
ORBIT_PARAMETERS	Orbit_Radius	Static	float64	Nominal distance in km from Earth's center to spacecraft track Valid format: NNNN.NNN, where NNNN.NNN = 7083.437	No	Yes	LMC System Spec
ORBIT_PARAMETERS	Orbit_Semimajor_Axis	Static	float64	Nominal semi-major axis in km of satellite's orbit Valid format: NNNN.NNNN, where NNNN.NNNN = 7077.9000	No	Yes	LMC System Spec
ORBIT_PARAMETERS	Orbit_Semiminor_Axis	Static	float64	Nominal semi-minor axis in km of satellite's orbit Valid format: NNNN.NNNN, where NNNN.NNNN = 7069.5800	No	Yes	LMC System Spec
ORBIT_PARAMETERS	Orbit_Eccentricity	Static	float64	Nominal eccentricity of satellite's orbit Valid format: N.NNNNN, where N.NNNNN = 0.00118	No	Yes	LMC System Spec
ORBIT_PARAMETERS	Inclination_Angle	Static	float64	Angle in degrees formed by Earth equatorial and satellite plane Valid format: NN.NNNN, where NN.NNNN = 98.2098	No	Yes	LMC System Spec
ORBIT_PARAMETERS	Argument_Of_Perigee	Static	float32	Nominal angle in degrees of point nearest Earth in orbit as measured from ascending node in direction of satellite motion Valid format: NN.N, where NN.N = 90.0	No	Yes	LMC System Spec
ORBIT_PARAMETERS	Descending_Node_ Row	Static	uint8	Row corresponding to Earth's equator Valid format: NN, where NN = 60	No	Yes	IAS
ORBIT_PARAMETERS	Long_Path1_Row60	Static	float32	Longitude in degrees west of point at which path 1 crossed equator (row 60) Valid format: SNN.N, where SNN.N = - 64.6	No	Yes	LMC System Spec
ORBIT_PARAMETERS	Descending_Node_ Time_Min	Static	char8	Minimum local solar time of descending node in AM hours and minutes Valid format: HH:MM, where HH:MM = 09:45	No	Yes	IAS
ORBIT_PARAMETERS	Descending_Node_ Time_Max	Static	char8	Maximum local solar time of descending node in AM hours and minutes Valid format: HH:MM, where HH:MM = 10:00	No	Yes	IAS
ORBIT_PARAMETERS	Nodal_Regression_ Rate	Static	float64	Rate in degrees per day that orbital plane rotates with respect to Earth Valid format: N.NNNNNNN, where N.NNNNNNN = 0.9856473	No	Yes	LMC System Spec

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
SCANNER_PARAMETERS	Lines_Per_Scan_30	Static	uint8	Detectors per scan for bands 1-5 and 7 Valid format: NN, where NN = 16	Yes	Yes	IAS
SCANNER_PARAMETERS	Lines_Per_Scan_60	Static	uint8	Detectors per scan for band 6 Valid format: N, where N = 8	Yes	Yes	IAS
SCANNER_PARAMETERS	Lines_Per_Scan_15	Static	uint8	Detectors per scan for band 8 Valid format: NN, where NN = 32	Yes	Yes	IAS
SCANNER_PARAMETERS	Scans_Per_Scene	Static	int16	Scans per nominal WRS scene Valid format: NNN, where NN = 375	Yes	Yes	IAS
SCANNER_PARAMETERS	Swath_Angle	Dynamic	float32	Object space angle in radians of scan mirror travel during active scan time Valid format: N.NNNNN, where N.NNNNN = 0.26868 (TBS after measurement of as-built ETM+)	No	Yes	SBRS
SCANNER_PARAMETERS	Scan_Rate	Static	float32	Angular scan velocity in radians per second of scan mirror Valid format: N.NNNNN, where N.NNNNN = 2.21095 (TBS)	No	Yes	SBRS
SCANNER_PARAMETERS	Dwell_Time_30	Static	float64	Detector sample time in microseconds for bands 1-5 and 7 Valid format: N.NNNNNNN, where N.NNNNNNN = 9.6109603	No	Yes	SBRS
SCANNER_PARAMETERS	Dwell_Time_60	Static	float64	Detector sample time in microseconds for band 6 Valid format: N.NNNNNNN, where N.NNN = 19.2220000	No	Yes	SBRS
SCANNER_PARAMETERS	Dwell_Time_15	Static	float64	Detector sample time in microseconds for band 8 Valid format: NN.NNNNNNN, where N.NNN = 4.8060000	No	Yes	SBRS
SCANNER_PARAMETERS	IC_Line_Length_30	Static	int16	Nominal number of detector samples for internal calibrator for bands 1-5 and 7 Valid format: NNNN, where NNNN = 1100	No	Yes	SBRS
SCANNER_PARAMETERS	IC_Line_Length_60	Static	int16	Nominal number of detector samples for internal calibrator for band 6 Valid format: NNNN, where NNNN = 550	No	Yes	SBRS
SCANNER_PARAMETERS	IC_Line_Length_15	Static	int16	Nominal number of detector samples for internal calibrator for band 8 Valid format: NNNNN, where NNNNN = 2200	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
SCANNER_PARAMETERS	Scan_Line_Length_30	Static	int16	Nominal number of detector samples during active scan time for bands 1-5 and 7 Valid format: NNNN, where NNNN = 6330	No	Yes	SBRS
SCANNER_PARAMETERS	Scan_Line_Length_60	Static	int16	Nominal number of detector samples during active scan time for band 6 Valid format: NNNN, where NNNN = 3165	No	Yes	SBRS
SCANNER_PARAMETERS	Scan_Line_Length_15	Static	int16	Nominal number of detector samples during active scan time for band 8 Valid format: NNNNN, where NNNNN = 12660	No	Yes	SBRS
SCANNER_PARAMETERS	Filter_Frequency_30	Static	float32	Bandwidth in kHz of detector presample filter (defined by 3-dB roll-off point) for bands 1-5 and 7 Valid format: NN.NN, where NN.NN = 52.02	No	Yes	SBRS
SCANNER_PARAMETERS	Filter_Frequency_60	Static	float32	Bandwidth in kHz of detector presample filter (defined by 3-dB roll-off point) for band 6 Valid format: NN.NN, where NN.NN = 26.01	No	Yes	SBRS
SCANNER_PARAMETERS	Filter_Frequency_15	Static	float32	Bandwidth in kHz of detector presample filter (defined by 3-dB roll-off point) for band 8 Valid format: NNN.NN, where NNN.NN = 115.00	No	Yes	SBRS
SCANNER_PARAMETERS	IFOV_B1234	Static	float32	Angle in µrad subtended by detector when scanning motion is stopped Valid format: NN.NNNN, where NN.NNNN = 42.5000 (TBS)	No	Yes	SBRS
SCANNER_PARAMETERS	IFOV_B57_along_ scan	Static	float32	Angle in µrad subtended by detector when scanning motion is stopped Valid format: NN.N, where NN.N = 39.6 (TBS)	No	Yes	SBRS
SCANNER_PARAMETERS	IFOV_B57_across_ scan	Static	float32	Angle in µrad subtended by detector when scanning motion is stopped Valid format: NN.N, where NN.N = 42.5 (TBS)	No	Yes	SBRS
SCANNER_PARAMETERS	IFOV_B6	Static	float32	Angle in µrad subtended by detector when scanning motion is stopped Valid format: NN.N, where NN.N = 85.0 (TBS)	No	Yes	SBRS
SCANNER_PARAMETERS	IFOV_B8_along_scan	Static	float32	Angle in µrad subtended by detector when scanning motion is stopped Valid format: NN.N, where NN.N = 18.5 (TBS)	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
SCANNER_PARAMETERS	IFOV_B8_across_scan	Static	float32	Angle in µrad subtended by detector when scanning motion is stopped Valid format: NN.N, where NN.N = 21.25 (TBS)	No	Yes	SBRS
SCANNER_PARAMETERS	Scan_Period	Static	float64	Time in milliseconds of a complete scan cycle, including forward and reverse scans Valid format: NNN.NNNNNN, where NNN.NNNNNN = 142.925000	No	Yes	SBRS
SCANNER_PARAMETERS	Scan_Frequency	Static	float32	Number of scans in 1 second (Hz) Valid format: N.NNNN, where N.NNNN = 6.9967	No	Yes	SBRS
SCANNER_PARAMETERS	Active_Scan_Time	Static	float32	Time in µs required for scan mirror to travel from its scan-line-start to end-of-line (EOL) Valid format: NNNNN.N, where NNNNN.N = 60743.0	No	Yes	SBRS
SCANNER_PARAMETERS	Turn_Around_Time	Static	float32	Time in milliseconds from EOL to next scan-line-start, during which scan mirror motion reverses direction Valid format: NN.NNN, where: NN.NNN = 10.719 (TBS)	No	Yes	SBRS
SPACECRAFT_PARAMETERS	ADS_Interval	Static	float32	Time in milliseconds between ADS samples Valid format: N.N, where N.N = 2.0	No	Yes	SBRS
SPACECRAFT_PARAMETERS	ADS_Roll_Offset	Static	float32	Amount of time in milliseconds from start of a payload correction data (PCD) cycle to roll axis measurement Valid format: N.NNN, where N.NNN = 0.375	No	Yes	SBRS
SPACECRAFT_PARAMETERS	ADS_Yaw_Offset	Static	float32	Amount of time in milliseconds from start of a PCD cycle to yaw axis measurement Valid format: N.NNN, where N.NNN = 0.875	No	Yes	SBRS
SPACECRAFT_PARAMETERS	ADS_Pitch_Offset	Static	float32	Amount of time in milliseconds from start of a PCD cycle to pitch axis measurement Valid format: N.NNN, where N.NNN = 1.375	No	Yes	SBRS
SPACECRAFT_PARAMETERS	Data_Rate	Static	float32	ETM+ output bit rate in Mbps Valid format: NN.NNN, where NN.NNN = 74.903 (TBS)	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
MIRROR_PARAMETERS GROUP: ANGLES_SME1_SAM	Forward_Along_ SME1_SAM	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe departure from linearity of forward along scan mirror motion; scan angle monitor (SAM) with scan mirror electronics (SME) number 1 Valid format: for each term: SN.NNNNESN, where S = “+” or “-“, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_SAM	Forward_Cross_ SME1_SAM	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of forward cross scan mirror motion from linear; SAM mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_SAM	Forward_Angle1_ SME1_SAM	Static	float32	Angle in µrad from start of scan to mid-scan point in forward direction; SAM mode Valid format: NNNNN.N, where NNNNN.N = 67171.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_SAM	Forward_Angle2_ SME1_SAM	Static	float32	Angle in µrad from mid-scan point to end of scan in forward direction; SAM mode Valid format: NNNNN.N, where NNNNN.N = 67159.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_SAM	Reverse_Along_ SME1_SAM	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of reverse along scan mirror motion from linear; SAM mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_SAM	Reverse_Cross_ SME1_SAM	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of reverse cross scan mirror motion from linear; SAM mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_SAM	Reverse_Angle1_ SME1_SAM	Static	float32	Angle in µrad from start of scan to mid-scan point in reverse direction; SAM mode Valid format: NNNNN.N, where NNNNN.N = 67159.0	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_SAM	Reverse_Angle2_ SME1_SAM	Static	float32	Angle in µrad from mid-scan point to end of scan in reverse direction; SAM mode Valid format: NNNNN.N, where NNNNN.N = 67171.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_SAM	Forward_Along_ SME2_SAM	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of forward along scan mirror motion from linear; SAM mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_SAM	Forward_Cross_SME2_ SAM	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of forward cross scan mirror motion from linear; SAM mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_SAM	Forward_Angle1_ SME2_SAM	Static	float32	Angle in µrad from start of scan to mid-scan point in forward direction; SAM mode Valid format: NNNNN.N, where NNNNN.N = 67182.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_SAM	Forward_Angle2_ SME2_SAM	Static	float32	Angle in µrad from mid-scan point to end of scan in forward direction; SAM mode Valid format: NNNNN.N, where NNNNN.N = 67160.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_SAM	Reverse_Along_ SME2_SAM	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of reverse along scan mirror motion from linear; SAM mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_SAM	Reverse_Cross_ SME2_SAM	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of reverse cross scan mirror motion from linear; SAM mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_SAM	Reverse_Angle1_ SME2_SAM	Static	float32	Angle in µrad from start of scan to mid-scan point in reverse direction; SAM mode Valid format: NNNNN.N, where NNNNN.N = 67160.0	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_SAM	Reverse_Angle2_ SME2_SAM	Static	float32	Angle in µrad from mid-scan point to end of scan in reverse direction; SAM mode Valid format: NNNNN.N, where NNNNN.N = 67182.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_BUMP	Forward_Along_ SME1_Bump	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of forward along scan mirror motion from linear; Bumper mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_BUMP	Forward_Cross_ SME1_Bump	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of forward cross scan mirror motion from linear; Bumper mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_BUMP	Forward_Angle1_ SME1_Bump	Static	float32	Angle in µrad from start of scan to mid-scan point in forward direction; Bumper mode Valid format: NNNNN.N, where NNNNN.N = 67171.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_BUMP	Forward_Angle2_ SME1_Bump	Static	float32	Angle in µrad from mid-scan point to end of scan in forward direction; Bumper mode Valid format: NNNNN.N, where NNNNN.N = 67159.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_BUMP	Reverse_Along_ SME1_Bump	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of reverse along scan mirror motion from linear; Bumper mode Valid format: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_BUMP	Reverse_Cross_ SME1_Bump	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of reverse cross scan mirror motion from linear; Bumper mode Valid format: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_BUMP	Reverse_Angle1_ SME1_Bump	Static	float32	Angle in µrad from start of scan to mid-scan point in reverse direction; Bumper mode Valid format: NNNNN.N, where NNNNN.N = 67159.0	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME1_BUMP	Reverse_Angle2_ SME1_Bump	Static	float32	Angle in µrad from mid-scan point to end of scan in reverse direction; Bumper mode Valid format: NNNNN.N, where NNNNN.N = 67171.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_BUMP	Forward_Along_ SME2_Bump	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of forward along scan mirror motion from linear; Bumper mode Valid format: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_BUMP	Forward_Cross_ SME2_Bump	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of forward cross scan mirror motion from linear; Bumper mode Valid format: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_BUMP	Forward_Angle1_ SME2_Bump	Static	float32	Angle in µrad from start of scan to mid-scan point in forward direction; Bumper mode Valid format: NNNNN.N, where NNNNN.N = 67182.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_BUMP	Forward_Angle2_ SME2_Bump	Static	float32	Angle in µrad from mid-scan point to end of scan in forward direction; Bumper mode Valid format: NNNNN.N, where NNNNN.N = 67162.0	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_BUMP	Reverse_Along_ SME2_Bump	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of reverse along scan mirror motion from linear; Bumper mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_BUMP	Reverse_Cross_ SME2_Bump	Static	float64 array (6 values)	Fifth-order polynomial coefficients that describe deviation of reverse cross scan mirror motion from linear; Bumper mode Valid format: for each term: SN.NNNNESN, where S = “+” or “-”, N = 0 to 9, and E = "E"	No	Yes	SBRS
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_BUMP	Reverse_Angle1_ SME2_Bump	Static	float32	Angle in µrad from start of scan to mid-scan point in reverse direction; Bumper mode Valid format: NNNNN.N, where NNNNN.N = 67160.0	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: MIRROR_PARAMETERS GROUP: ANGLES_SME2_BUMP	Reverse_Angle2_ SME2_Bump	Static	float32	Angle in µrad from mid-scan point to end of scan in reverse direction; Bumper mode Valid format: NNNNN.N, where NNNNN.N = 67182.0	No	Yes	SBRS
GROUP: MIRROR_ PARAMETERS	Error_Conversion_ Factor	Static	float32	First half and second half scan mirror error measurement units in microseconds Valid format. = N.NNNNN, where N.NNNNN = 0.18845 (5.306437 MHz)	No	Yes	SBRS
GROUP: SCAN_LINE_ CORRECTOR	Primary_Angular_ Velocity	Static	float32	Angular velocity in radians per second of primary scan line corrector Valid format: N.NNNNN, where N.NNNNN = 0.00966	No	Yes	SBRS
GROUP: SCAN_LINE_CORRECTOR	Secondary_Angular_ Velocity	Static	float32	Angular velocity in radians per second of secondary scan line corrector Valid format: N.NNNNN, where N.NNNNN = 0.00960	No	Yes	SBRS
GROUP: SCAN_LINE_CORRECTOR	Primary_Corrector_ Motion	Static	float32 array (6 values)	Fifth-order polynomial coefficients that describe motion of primary scan line corrector Valid format: for each term: N.NNNNN, where N.NNNNN = 0.0 (TBS)	No	Yes	SBRS
GROUP: SCAN_LINE_CORRECTOR	Secondary_ Corrector_Motion	Static	float32 array (6 values)	Fifth-order polynomial coefficients that describe motion of secondary scan line corrector Valid format: for each term: N.NNNNN, where N.NNNNN = 0.0 (TBS)	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: BAND_OFFSETS	Along_Scan_Band_ Offsets	Static	float32 array (8 values)	Nominal displacement in µrad from center of focal plane to each band's optical axis Valid format: NNNN.NNN, where NNNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: BAND_OFFSETS	Across_Scan_Band_ Offsets	Static	float32 array (8 values)	Nominal displacement in µrad from center of focal plane to each band's scan motion axis Valid format: NNNN.NNN, where NNNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: BAND_OFFSETS	Forward_Focal_ Plane_Offsets	Static	float32 array (8 values)	Offset in instrument fields of view (IFOVs) for focal plane forward scans Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: BAND_OFFSETS	Reverse_Focal_ Plane_Offsets	Static	float32 array (8 values)	Offset in IFOVs for focal plane reverse scans Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Along_Scan_DO_B1	Static	float32 array (16 values)	Forward along scan detector offsets in IFOV for each detector in band 1 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Along_Scan_DO_B1	Static	float32 array (16 values)	Reverse along scan detector offsets in IFOV for each detector in band 1 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Along_Scan_DO_B2	Static	float32 array (16 values)	Forward along scan detector offsets in IFOV for each detector in band 2 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Along_Scan_DO_B2	Static	float32 array (16 values)	Reverse along scan detector offsets in IFOV for each detector in band 2 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Along_Scan_DO_B3	Static	float32 array (16 values)	Forward along scan detector offsets in IFOV for each detector in band 3 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Along_Scan_DO_B3	Static	float32 array (16 values)	Reverse along scan detector offsets in IFOV for each detector in band 3 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Along_Scan_DO_B4	Static	float32 array (16 values)	Forward along scan detector offsets in IFOV for each detector in band 4 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Along_Scan_DO_B4	Static	float32 array (16 values)	Reverse along scan detector offsets in IFOV for each detector in band 4 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Along_Scan_DO_B5	Static	float32 array (16 values)	Forward along scan detector offsets in IFOV for each detector in band 5 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Along_Scan_DO_B5	Static	float32 array (16 values)	Reverse along scan detector offsets in IFOV for each detector in band 5 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Along_Scan_DO_B6	Static	float32 array (8 values)	Forward along scan detector offsets in IFOV for each detector in band 6 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Along_Scan_DO_B6	Static	float32 array (8 values)	Reverse along scan detector offsets in IFOV for each detector in band 6 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Along_Scan_DO_B7	Static	float32 array (16 values)	Forward along scan detector offsets in IFOV for each detector in band 7 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Along_Scan_DO_B7	Static	float32 array (16 values)	Reverse along scan detector offsets in IFOV for each detector in band 7 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Along_Scan_DO_B8	Static	float32 array (32 values)	Forward along scan detector offsets in IFOV for each detector in band 8 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Along_Scan_DO_B8	Static	float32 array (32 values)	Reverse along scan detector offsets in IFOV for each detector in band 8 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Across_Scan_DO_B1	Static	float32 array (16 values)	Forward across scan detector offsets in IFOV for each detector in band 1 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Across_Scan_DO_B1	Static	float32 array (16 values)	Reverse across scan detector offsets in IFOV for each detector in band 1 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Across_Scan_DO_B2	Static	float32 array (16 values)	Forward across scan detector offsets in IFOV for each detector in band 2 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Across_Scan_DO_B2	Static	float32 array (16 values)	Reverse across scan detector offsets in IFOV for each detector in band 2 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Across_Scan_DO_B3	Static	float32 array (16 values)	Forward across scan detector offsets in IFOV for each detector in band 3 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Across_Scan_DO_B3	Static	float32 array (16 values)	Reverse across scan detector offsets in IFOV for each detector in band 3 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Across_Scan_DO_B4	Static	float32 array (16 values)	Forward across scan detector offsets in IFOV for each detector in band 4 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Across_Scan_DO_B4	Static	float32 array (16 values)	Reverse across scan detector offsets in IFOV for each detector in band 4 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Across_Scan_DO_B5	Static	float32 array (16 values)	Forward across scan detector offsets in IFOV for each detector in band 5 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Across_Scan_DO_B5	Static	float32 array (16 values)	Reverse across scan detector offsets in IFOV for each detector in band 5 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Across_Scan_DO_B6	Static	float32 array (8 values)	Forward across scan detector offsets in IFOV for each detector in band 6 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Across_Scan_DO_B6	Static	float32 array (8 values)	Reverse across scan detector offsets in IFOV for each detector in band 6 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Across_Scan_DO_B7	Static	float32 array (16 values)	Forward across scan detector offsets in IFOV for each detector in band 7 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Across_Scan_DO_B7	Static	float32 array (16 values)	Reverse across scan detector offsets in IFOV for each detector in band 7 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Forward_Across_Scan_DO_B8	Static	float32 array (32 values)	Forward across scan detector offsets in IFOV for each detector in band 8 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: DETECTOR_OFFSETS	Reverse_Across_Scan_DO_B8	Static	float32 array (32 values)	Reverse across scan detector offsets in IFOV for each detector in band 8 Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: ODD_EVEN_OFFSETS	Forward_Even_Detector_Shift	Static	float32 array (8 values)	Adjustments in IFOVs to compensate for forward band offsets, even detector layout geometry and multiplexer sampling for bands 1 through 8 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: FOCAL_PLANE_PARAMETERS GROUP: ODD_EVEN_OFFSETS	Forward_Odd_ Detector_Shift	Static	float32 array (8 values)	Adjustments in IFOVs to compensate for forward band offsets, odd detector layout geometry and multiplexer sampling for bands 1 through 8 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: ODD_EVEN_OFFSETS	Reverse_Even_Detector_ Shift	Static	float32 array (8 values)	Adjustments in IFOVs to compensate for reverse band offsets, even detector layout geometry and multiplexer sampling for bands 1 through 8 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	Yes	SBRS
GROUP: FOCAL_PLANE_PARAMETERS GROUP: ODD_EVEN_OFFSETS	Reverse_Odd_Detector_ Shift	Static	float32 array (8 values)	Adjustments in IFOVs to compensate for reverse band offsets, odd detector layout geometry and multiplexer sampling for bands 1 through 8 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	Yes	SBRS
GROUP: ATTITUDE_PARAMETERS	Gyro_To_Attitude_ Matrix	Static	float32 array (9 values)	Matrix describing relationship of gyro axis to attitude control reference axis Valid format: N.N, where N.N = (1.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 1.0)	Yes	Yes	LMC
GROUP: ATTITUDE_PARAMETERS	ADSA_To_ETM_Matrix	Static	float32 array (9 values)	Matrix describing relationship of ADSA to ETM+ optical Axis Valid format: N.N, where N.N = (1.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 1.0)	Yes	Yes	LMC
GROUP: ATTITUDE_PARAMETERS	Attitude_To_ETM_ Matrix	Static	float32 array (9 values)	Matrix describing relationship of attitude control reference axis to ETM+ optical axis Valid format: N.N, where N.N = (1.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 1.0)	Yes	Yes	LMC
GROUP: ATTITUDE_PARAMETERS	Spacecraft_Roll_Bias	Static	float32	Spacecraft roll bias in radians Valid format: N.NNNNNNN, where N.NNNNNNN = TBS	Yes	Yes	LMC
GROUP: ATTITUDE_PARAMETERS	Spacecraft_Pitch_Bias	Static	float32	Spacecraft pitch bias in radians Valid format: N.NNNNNNN, where N.NNNNNNN = TBS	Yes	Yes	LMC
GROUP: ATTITUDE_PARAMETERS	Spacecraft_Yaw_Bias	Static	float32	Spacecraft yaw bias in radians Valid format: N.NNNNNNN, where N.NNNNNNN = TBS	Yes	Yes	LMC
GROUP: TIME_PARAMETERS	Scan_Time	Static	float32	Nominal scan time in microseconds Valid format: NNNNN.N, where NNNNN.N = 60743.0	No	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: TIME_PARAMETERS	Forward_First_Half_ Time	Static	float32	Nominal forward first half scan time in microseconds Valid format: NNNNN.N, where NNNNN.N = 30371.4	No	Yes	SBRS
GROUP: TIME_PARAMETERS	Forward_Second_Half_ Time	Static	float32	Nominal forward second half scan time in microseconds Valid format: NNNNN.N, where NNNNN.N = 30371.6	No	Yes	SBRS
GROUP: TIME_PARAMETERS	Reverse_First_Half_ Time	Static	float32	Nominal reverse first half scan time in microseconds Valid format: NNNNN.N, where NNNNN.N = 30371.6	No	Yes	SBRS
GROUP: TIME_PARAMETERS	Reverse_Second_ Half_Time	Static	float32	Nominal reverse second half scan time in microseconds Valid format: NNNNN.N, where NNNNN.N = 30371.4	No	Yes	SBRS
GROUP: TRANSFER_FUNCTION GROUP: IMU	Fn	Static	float64	Inertial measurement unit transfer function resonant frequency (Hz) Valid format: N.NNNN, where N.NNNN = TBS	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: IMU	Zeta	Static	float64	Inertial measurement unit transfer function damping coefficient Valid format: N.NNNN, where N.NNNN = TBS	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: IMU	Tau	Static	float64	Inertial measurement unit transfer function denominator time constant (seconds) Valid format: NN.NNNNENN, where NN.NNNNENN = TBS	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: IMU	P	Static	float64	Inertial measurement unit transfer function numerator time constant (seconds) Valid format: NN.NNNNENN, where NN.NNNNENN = TBS	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: IMU	Ak	Static	float64	Inertial measurement unit transfer function DC gain Valid format: N.NNNNN, where N.NNNNN = TBS	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: ADS	ADS_num	Static	float64 array (18 values)	Transfer function numerator coefficients in order a0, a1, a2, a3, a4, a5; one set of six coefficients for each of three ADS units; determined at 15 degrees C Valid format: N.NNNNEN, where: N.NNNNEN = TBS	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: ADS	ADS_den	Static	float64 array (18 values)	Transfer function denominator coefficients in order b0, b1, b2, b3, b4, b5; one set of six coefficients for each of three ADS units; determined at 15 degrees C Valid format: N.NNNNEN, where: N.NNNNEN = TBS	No	Yes	LMC

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: TRANSFER_FUNCTION GROUP: ADS	ADS_num_temp	Static	float64 array (18 values)	Temperature dependent part of ADS transfer function numerator coefficients in order da0, da1, da2, da3, da4, da5; one set of six coefficients for each of three ADS units; change per degree C Valid format: N.NNNNEN, where: N.NNNNEN = TBS	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: ADS	ADS_den_temp	Static	float64 array (18 values)	Temperature dependent part of ADS transfer function denominator coefficients in order da0, da1, da2, da3, da4, da5. One set of six coefficients for each of three ADS units. Change per degree C Valid format: N.NNNNEN, where: N.NNNNEN = TBS	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: PREFILTER	ADSPre_W	Static	float64 array (5 values)	ADS prefilter transfer function quadratic term resonant periods (Note: Given as period instead of frequency so that the transfer function can be set to unity, if necessary, by setting all five values to zero.) Valid format: N.N, where N.N = 0.0	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: PREFILTER	ADSPre_H	Static	float64 array (5 values)	ADS prefilter transfer function quadratic term damping coefficients Valid format: N.N, where N.N = 0.0	No	Yes	LMC
GROUP: TRANSFER_FUNCTION GROUP: PREFILTER	ADSPre_T	Static	float64 array (5 values)	ADS prefilter transfer function linear term time constants Valid format: N.N, where N.N = 0.0	No	Yes	LMC
GROUP: UT1_TIME_PARAMETERS	UT1_Year	Dynamic	int16 array (180 values)	Year of UT1 time correction prediction; values span 180 days Valid format: YYYY, where YYYY = 1998-2008	Yes	Yes	Natl. Earth Orientation Service
GROUP: UT1_TIME_PARAMETERS	UT1_Month	Dynamic	char8 array (180 values)	Month of UT1 time correction prediction; values span 180 days Valid format: MMM, where MMM = Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, or Dec	Yes	Yes	NEOS
GROUP: UT1_TIME_PARAMETERS	UT1_Day	Dynamic	uint8 array (180 values)	Day of UT1 time correction prediction; values span 180 days Valid format: NN, where NN = 1-31	Yes	Yes	NEOS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: UT1_TIME_PARAMETERS	UT1_Modified_Julian	Dynamic	int32 array (180 values)	Modified Julian day; values span 180 days; MJD = Julian day - 2 400 000.5; Julian date is a running day count starting 1 January 4713 B.C Valid format: NNNNN, where NNNNN = 50234 (May 31, 1996)	Yes	Yes	NEOS
GROUP: UT1_TIME_PARAMETERS	UT1_X	Dynamic	float32 array (180 values)	X shift pole wander in arc seconds; values span 180 days Valid format: N.NNNNN, where N.NNNNN = e.g. 0.45431	Yes	Yes	NEOS
GROUP: UT1_TIME_PARAMETERS	UT1_Y	Dynamic	float32 array (180 values)	Y shift pole wander in arc seconds; values span 180 days Valid format: N.NNNNN, where N.NNNNN = e.g., 0.13454	Yes	Yes	NEOS
GROUP: UT1_TIME_PARAMETERS	UT1_UTC	Dynamic	float32 array (180 values)	UT1 - UTC time difference in seconds. Values span 180 days Valid format: N.NNNNN, where N.NNNNN = (e.g., 0.44321)	Yes	Yes	NEOS
GROUP: DETECTOR_STATUS	Status_Band1	Dynamic	char8 array (16 values)	Health status of band 1's 16 detectors Valid format: ABCDE, where A = 0 (live), 1 (dead), 2 (intermittent) B = 0 (noise in spec, low gain), 1 (noisy low signal), 2 (noisy high signal), 3(noisy both signals) C = 0 (noise in spec, high gain), 1 (noisy low signal), 2 (noisy high signal), 3(noisy both signals) D = 0 (dynamic range in spec, low gain) 1 (fail, high end), 2 (fail, low end), 3 (fail, both ends) E = 0 (dynamic range in spec, high gain), 1 (fail, low end), 2 (fail, low end), 3 (fail, both ends)	No	Yes	AC02/ AC48
GROUP: DETECTOR_STATUS	Status_Band2	Dynamic	char8 array (16 values)	Health status of band 2's 16 detectors Valid format: as above	No	Yes	AC02/ AC48
GROUP: DETECTOR_STATUS	Status_Band3	Dynamic	char8 array (16 values)	Health status of band 3's 16 detectors Valid format: as above.	No	Yes	AC02/ AC48
GROUP: DETECTOR_STATUS	Status_Band4	Dynamic	char8 array (16 values)	Health status of band 4's 16 detectors Valid format: as above	No	Yes	AC02/ AC48
GROUP: DETECTOR_STATUS	Status_Band5	Dynamic	char8 array (16 values)	Health status of band 5's 16 detectors Valid format: as above	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_STATUS	Status_Band6	Dynamic	char8 array (8 values)	Health status of band 6's 8 detectors Valid format: as above	No	Yes	BL10 in thermal vacuum
GROUP: DETECTOR_STATUS	Status_Band7	Dynamic	char8 array (16 values)	Health status of band 7's 16 detectors Valid format: as above	No	Yes	AC02/ AC48
GROUP: DETECTOR_STATUS	Status_Band8	Dynamic	char8 array (32 values)	Health status of band 8's 32 detectors Valid format: as above	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B1L_Prelaunch	Static	float32 array (16 values)	Band 1 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B1L_Postlaunch	Static	float32 array (16 values)	Band 1 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B1L_Current	Dynamic	float32 array (16 values)	Band 1 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B2L_Prelaunch	Static	float32 array (16 values)	Band 2 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B2L_Postlaunch	Static	float32 array (16 values)	Band 2 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B2L_Current	Dynamic	float32 array (16 values)	Band 2 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B3L_Prelaunch	Static	float32 array (16 values)	Band 3 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B3L_Postlaunch	Static	float32 array (16 values)	Band 3 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B3L_Current	Dynamic	float32 array (16 values)	Band 3 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B4L_Prelaunch	Static	float32 array (16 values)	Band 4 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B4L_Postlaunch	Static	float32 array (16 values)	Band 4 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B4L_Current	Dynamic	float32 array (16 values)	Band 4 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B5L_Prelaunch	Static	float32 array(16 values)	Band 5 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B5L_Postlaunch	Static	float32 array (16 values)	Band 5 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B5L_Current	Dynamic	float32 array (16 values)	Band 5 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B6L_Prelaunch	Static	float32 array (8 values)	Band 6 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	BL10 in thermal vacuum
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B6L_Postlaunch	Static	float32 array (8 values)	Band 6 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B6L_Current	Dynamic	float32 array (8 values)	Band 6 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	BL10 in thermal vacuum
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B7L_Prelaunch	Static	float32 array (16 values)	Band 7 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B7L_Postlaunch	Static	float32 array (16 values)	Band 7 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B7L_Current	Dynamic	float32 array (16 values)	Band 7 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B8L_Prelaunch	Static	float32 array (32 values)	Band 8 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B8L_Postlaunch	Static	float32 array (32 values)	Band 8 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_LOW	B8L_Current	Dynamic	float32 array (32 values)	Band 8 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B1H_Prelaunch	Static	float32 array (16 values)	Band 1 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B1H_Postlaunch	Static	float32 array (16 values)	Band 1 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B1H_Current	Dynamic	float32 array (16 values)	Band 1 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B2H_Prelaunch	Static	float32 array (16 values)	Band 2 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B2H_Postlaunch	Static	float32 array (16 values)	Band 2 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B2H_Current	Dynamic	float32 array (16 values)	Band 2 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B3H_Prelaunch	Static	float32 array (16 values)	Band 3 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B3H_Postlaunch	Static	float32 array (16 values)	Band 3 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B3H_Current	Dynamic	float32 array (16 values)	Band 3 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B4H_Prelaunch	Static	float32 array (16 values)	Band 4 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B4H_Postlaunch	Static	float32 array (16 values)	Band 4 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B4H_Current	Dynamic	float32 array (16 values)	Band 4 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B5H_Prelaunch	Static	float32 array (16 values)	Band 5 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B5H_Postlaunch	Static	float32 array (16 values)	Band 5 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B5H_Current	Dynamic	float32 array (16 values)	Band 5 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B6H_Prelaunch	Static	float32 array (8 values)	Band 6 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	BL10 in thermal vacuum
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B6H_Postlaunch	Static	float32 array (8 values)	Band 6 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B6H_Current	Dynamic	float32 array (8 values)	Band 6 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	BL10 in thermal vacuum
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B7H_Prelaunch	Static	float32 array (16 values)	Band 7 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B7H_Postlaunch	Static	float32 array (16 values)	Band 7 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B7H_Current	Dynamic	float32 array (16 values)	Band 7 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B8H_Prelaunch	Static	float32 array (32 values)	Band 8 prelaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	AC02/ AC48
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B8H_Postlaunch	Static	float32 array (32 values)	Band 8 postlaunch gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_GAINS GROUP: DETECTOR_GAINS_HIGH	B8H_Current	Dynamic	float32 array (32 values)	Band 8 current gain in counts/W/m^2-ster-µm Valid format: NNN.NNNN, where NNN.NNNN = TBS	Yes	Yes	AC02/ AC48
GROUP: BIAS_LOCATIONS	Forward_Bias_Location_30	Dynamic	int16	Offset, per-line, in pixels, from beginning of data (Left Hand Offset) to bias location starting point (start of DC Restore) for bands 1–5 and 7 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Forward_Bias_Length_30	Dynamic	int16	Number of pixels to use, per line, in calculating bias for bands 1-5 and 7 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Forward_IC_Region_30	Dynamic	int16	Length of useable IC region, in pixels, from the start of the bias region (DC Restore) to the end of the calibration pulse region for bands 1-5 and 7 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Reverse_Bias_ Location_30	Dynamic	int16	Offset, per line, in pixels, from beginning of data (Right Hand Offset) to bias location starting point (start of DC Restore) for bands 1–5 and 7 Valid format: NNN, where NNN = TBS	No	Yes	IAS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: BIAS_LOCATIONS	Reverse_Bias_ Length_30	Dynamic	int16	Number of pixels to use per line, in calculating bias for bands 1-5 and 7 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Reverse_IC_ Region_30	Dynamic	int16	Length of useable IC region, in pixels, from the start of the bias region (DC Restore) to the end of the calibration pulse region for bands 1-5 and 7 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Forward_Bias_ Location_60	Dynamic	int16	Offset, per-line, in pixels, from beginning of data (Left Hand Offset) to bias location starting point (start of DC Restore) for band 6 Valid format: NNN, where NNN = TBS	No	Yes	LPSO – from pre-launch BL10
GROUP: BIAS_LOCATIONS	Forward_Bias_ Length_60	Dynamic	int16	Number of pixels to use, per line, in calculating bias for band 6 Valid format: NNN, where NNN = TBS	No	Yes	LPSO – from pre-launch BL10
GROUP: BIAS_LOCATIONS	Forward_IC_ Region_60	Dynamic	int16	Length of useable IC region, in pixels, from the start of the bias region (DC Restore) to the end of the calibration pulse region for band 6 Valid format: NNN, where NNN = TBS	No	Yes	LPSO – from pre-launch BL10
GROUP: BIAS_LOCATIONS	Reverse_Bias_ Location_60	Dynamic	int16	Offset, per line, in pixels, from beginning of data (Right Hand Offset) to bias location starting point (start of DC Restore) for band 6 Valid format: NNN, where NNN = TBS	No	Yes	LPSO – from pre-launch BL10
GROUP: BIAS_LOCATIONS	Reverse_Bias_ Length_60	Dynamic	int16	Number of pixels to use, per line, in calculating bias for band 6 Valid format: NNN, where NNN = TBS	No	Yes	LPSO – from pre-launch BL10
GROUP: BIAS_LOCATIONS	Reverse_IC_ Region_60	Dynamic	int16	Length of useable IC region, in pixels, from the start of the bias region (DC Restore) to the end of the calibration pulse region for band 6 Valid format: NNN, where NNN = TBS	No	Yes	LPSO – from pre-launch BL10

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: BIAS_LOCATIONS	Forward_Bias_ Location_15	Dynamic	int16	Offset, per-line, in pixels, from beginning of data (Left Hand Offset) to bias location starting point (start of DC Restore) for band 8 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Forward_Bias_ Length_15	Dynamic	int16	Number of pixels to use, per line, in calculating bias for band 8 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Forward_IC_ Region_15	Dynamic	int16	Length of useable IC region, in pixels, from the start of the bias region (DC Restore) to the end of the calibration pulse region for band 8 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Reverse_Bias_ Location_15	Dynamic	int16	Offset, per line, in pixels, from beginning of data (Right Hand Offset) to bias location starting point (start of DC Restore) for band 8 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Reverse_Bias_ Length_15	Dynamic	int16	Number of pixels to use, per line, in calculating bias for band 8 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: BIAS_LOCATIONS	Reverse_IC_ Region_15	Dynamic	int16	Length of useable IC region, in pixels, from the start of the bias region (DC Restore) to the end of the calibration pulse region for band 8 Valid format: NNN, where NNN = TBS	No	Yes	IAS
GROUP: DETECTOR_BIASES_B6 GROUP: DETECTOR_ BIASES_B6_LOW	B6L_Bias_Prelaunch	Static	float32 array (8 values)	Band 6 prelaunch low gain bias in digital counts Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	BL10 in thermal vacuum
GROUP: DETECTOR_BIASES_B6 GROUP: DETECTOR_ BIASES_B6_LOW	B6L_Bias_Postlaunch	Static	float32 array (8 values)	Band 6 postlaunch low gain bias in digital counts Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_BIASES_B6 GROUP: DETECTOR_ BIASES_B6_LOW	B6L_Bias_Current	Dynamic	float32 array (8 values)	Band 6 current low gain bias in digital counts Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	BL10 in thermal vacuum
GROUP: DETECTOR_BIASES_B6 GROUP: DETECTOR_ BIASES_B6_HIGH	B6H_Bias_Prelaunch	Static	float32 array (8 values)	Band 6 prelaunch high gain bias in digital counts Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	BL10 in thermal vacuum

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_BIASES_B6 GROUP: DETECTOR_BIASES_B6_HIGH	B6H_Bias_Postlaunch	Static	float32 array (8 values)	Band 6 postlaunch high gain bias in digital counts Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: DETECTOR_BIASES_B6 GROUP: DETECTOR_BIASES_B6_HIGH	B6H_Bias_Current	Dynamic	float32 array (8 values)	Band 6 current high gain bias in digital counts Valid format: NNN.NNNN, where NNN.NNNN = TBS	No	Yes	BL10 in thermal vacuum
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_LOW	B1L_ACCA_Bias	Dynamic	float32 array (16 values)	Band 1 low-gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_LOW	B2L_ACCA_Bias	Dynamic	float32 array (16 values)	Band 2 low-gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_LOW	B3L_ACCA_Bias	Dynamic	float32 array (16 values)	Band 3 low-gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_LOW	B4L_ACCA_Bias	Dynamic	float32 array (16 values)	Band 4 low-gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_LOW	B5L_ACCA_Bias	Dynamic	float32 array (16 values)	Band 5 low-gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_LOW	B6L_ACCA_Bias	Dynamic	float32 array (8 values)	Band 6 low-gain ACCA bias in digital counts for detectors 1-8 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_LOW	B7L_ACCA_Bias	Dynamic	float32 array (16 values)	Band 7 low-gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_LOW	B8L_ACCA_Bias	Dynamic	float32 array (32 values)	Band 8 low-gain ACCA bias in digital counts for detectors 1-32 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_HIGH	B1H_ACCA_Bias	Dynamic	float32 array (16 values)	Band 1 high gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_HIGH	B2H_ACCA_Bias	Dynamic	float32 array (16 values)	Band 2 high gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_HIGH	B3H_ACCA_Bias	Dynamic	float32 array (16 values)	Band 3 high gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_HIGH	B4H_ACCA_Bias	Dynamic	float32 array (16 values)	Band 4 high gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_HIGH	B5H_ACCA_Bias	Dynamic	float32 array (16 values)	Band 5 high gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_HIGH	B6H_ACCA_Bias	Dynamic	float32 array (8 values)	Band 6 high gain ACCA bias in digital counts for detectors 1- 8 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_HIGH	B7H_ACCA_Bias	Dynamic	float32 array (16 values)	Band 7 high gain ACCA bias in digital counts for detectors 1-16 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_BIASES GROUP: ACCA_BIASES_HIGH	B8H_ACCA_Bias	Dynamic	float32 array (32 values)	Band 8 high gain ACCA bias in digital counts for detectors 1-32 Valid format: NNN.NNN, where NNN.NNN = TBS	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_B3	Dynamic	float32	Band 3 ACCA threshold Valid format: N.NNNN, where N.NNNN = 0.3000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_B3_Lower	Dynamic	float32	Band 3 land reflectance threshold Valid format: NN.NN, where NN.NN = 0.06	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_B56_High	Dynamic	float32	Band 5-6 high composite threshold Valid format: NNN.NNN, where NNN.NNN = 225.000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_B56_Low	Dynamic	float32	Band 5-6 low composite threshold Valid format: NNN.NNN, where NNN.NNN = 210.000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_B6	Dynamic	float32	Band 6 threshold - maximum cloud temperature Valid format: NNN.NNN, where NNN.NNN = 300.000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_B45_Ratio	Dynamic	float32	Band 4-5 ratio threshold Valid format: N.NNNN, where N.NNNN = 1.0750	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_B42_Ratio	Dynamic	float32	Band 4-2 ratio threshold Valid format: N.NNNN, where N.NNNN = 2.0000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_B43_Ratio	Dynamic	float32	Band 4-3 ratio threshold Valid format: N.NNNN, where N.NNNN = 2.0000	Yes	No	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: ACCA_THRESHOLDS	Thresh_NDSI_Max	Dynamic	float32	Normalized snow difference index ceiling Valid format: N.NNNN, where N.NNNN = 0.7000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_NDSI_Min	Dynamic	float32	Normalized snow difference index floor Valid format: N.NNNN, where N.NNNN = -0.2500	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_NDSI_Snow	Dynamic	float32	NDSI threshold used to identify snow Valid format: NN.NNNN, where NN.NNNN = 0.8000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Cloud_Percent_Min	Dynamic	float32	Minimum cloud cover percentage required for pass two Valid format: N.NNNN, where N.NNNN = 0.4000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Desert_Index	Dynamic	float32	Desert index (Thresh_45_Ratio/ Thresh_42_Ratio) Valid format: N.NNNN, where N.NNNN = 0.5000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thresh_Snow_Percent	Dynamic	float32	Maximum snow cover percentage allowed to use looser cloud properties for pass two Valid format: N.NNN, where N.NNN = 1.000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thermal_Effect_High	Dynamic	float32	Maximum allowable pass two percentage cloud cover increase allowed using looser cloud properties Valid format: NNN.NNN, where NNN.NNN = 40.000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	Thermal_Effect_Low	Dynamic	float32	Maximum allowable pass two percentage cloud cover increase allowed using narrower cloud properties Valid format: NNN.NNN, where NNN.NNN = 30.000	Yes	No	LPSO
GROUP: ACCA_THRESHOLDS	B6Max_Maxthresh_Diff	Dynamic	float32	Minimum difference allowed between maximum cloud temperature and maximum thermal threshold Valid format: NN.NNN, where NN.NNN = 2.000	Yes	No	LPSO
GROUP: SOLAR_SPECTRAL_ IRRADIANCES	B1_Solar_Irradiance	Static	float32	Mean solar exoatmospheric irradiance for band 1 in W/m^2-ster-µm Valid format: NNNN.NNN, where NNNN.NNN = 1957.000	Yes	No	LPSO
GROUP: SOLAR_SPECTRAL_ IRRADIANCES	B2_Solar_Irradiance	Static	float32	Mean solar expatmosphere irradiance for band 2 in W/m^2-ster-µm Valid format: NNNN.NNN, where NNNN.NNN = 1829.000	Yes	No	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SOLAR_SPECTRAL_ IRRADIANCES	B3_Solar_Irradiance	Static	float32	Mean solar expatmosphere irradiance for band 3 in W/m^2-ster-µm Valid format: NNNN.NNN, where NNNN.NNN = 1557.000	Yes	No	LPSO
GROUP: SOLAR_SPECTRAL_ IRRADIANCES	B4_Solar_Irradiance	Static	float32	Mean solar expatmosphere irradiance for band 4 in W/m^2-ster-µm Valid format: NNNN.NNN, where NNNN.NNN = 1047.000	Yes	No	LPSO
GROUP: SOLAR_SPECTRAL_ IRRADIANCES	B5_Solar_Irradiance	Static	float32	Mean solar expatmosphere irradiance for band 5 in W/m^2-ster-µm Valid format: NNNN.NNN, where NNNN.NNN = 219.300	Yes	No	LPSO
GROUP: SOLAR_SPECTRAL_ IRRADIANCES	B7_Solar_Irradiance	Static	float32	Mean solar expatmosphere irradiance for band 7 in W/m^2-ster-µm Valid format: NNNN.NNN, where NNNN.NNN = 74.520	Yes	No	LPSO
GROUP: SOLAR_SPECTRAL_ IRRADIANCES	B8_Solar_Irradiance	Static	float32	Mean solar expatmosphere irradiance for band 8 in W/m^2-ster-µm Valid format: NNNN.NNN, where NNNN.NNN = TBS	Yes	No	LPSO
GROUP: THERMAL_CONSTANTS	K1_Constant	Static	float32	Thermal calibration constant 1 in W/m^2-ster-µm Valid format: NNNNN.NNN, where NNNNN.NNN = 607.760	Yes	No	LPSO
GROUP: THERMAL_CONSTANTS	K2_Constant	Static	float32	Thermal calibration constant 2 in degrees kelvin Valid format: NNNNN.NNN, where NNNNN.NNN = 1260.560	Yes	No	LPSO
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS _LOW	B1L_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 1, low gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS _LOW	B2L_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 2, low gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS _LOW	B3L_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 3, low gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS_LOW	B4L_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 4, low gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS_LOW	B5L_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 5, low gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS_LOW	B6L_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 6, low gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS_LOW	B7L_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 7, low gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS_LOW	B8L_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 8, low gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS_HIGH	B1H_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 1, high gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS_HIGH	B2H_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 2, high gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_PARAMETERS_HIGH	B3H_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 3, high gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SCALING_PARAMETERS GROUP: SCALING_ PARAMETERS_HIGH	B4H_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 4, high gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_ PARAMETERS_HIGH	B5H_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 5, high gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_ PARAMETERS_HIGH	B6H_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 6, high gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_ PARAMETERS_HIGH	B7H_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 7, high gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: SCALING_PARAMETERS GROUP: SCALING_ PARAMETERS_HIGH	B8H_Lmin_Lmax	Static	float32 array (2 values)	Postcalibration 8-bit dynamic range scaling factors for band 8, high gain, W/m^2-ster-µm Valid format: SNN.NNN, where S = "+" or "-" and NN.NNN = TBS	Yes	Yes	Postlaunch
GROUP: MTF_COMPENSATION	B1_weights_along	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute along scan MTFC for band 1 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B1_weights_across	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute across scan MTFC for band 1 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B2_weights_along	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute along scan MTFC for band 2 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B2_weights_across	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute across scan MTFC for band 2 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: MTF_COMPENSATION	B3_weights_along	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute along scan MTFC for band 3 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B3_weights_across	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute across scan MTFC for band 3 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B4_weights_along	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute along scan MTFC for band 4 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B4_weights_across	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute across scan MTFC for band 4 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B5_weights_along	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute along scan MTFC for band 5 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B5_weights_across	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute across scan MTFC for band 5 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B6_weights_along	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute along scan MTFC for band 6 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B6_weights_across	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute across scan MTFC for band 6 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B7_weights_along	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute along scan MTFC for band 7 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: MTF_COMPENSATION	B7_weights_across	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute across scan MTFC for band 7 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B8_weights_along	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute along scan MTFC for band 8 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MTF_COMPENSATION	B8_weights_across	Dynamic	float64 array (5 values)	Weighting function coefficients used to compute across scan MTFC for band 8 Valid format: SNN.NNNN, where S = "+" or "-" and NN.NNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_MAGNITUDES	B1_ME_Magnitude	Dynamic	float32 array (16 values)	Band 1 memory effect magnitude measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_MAGNITUDES	B2_ME_Magnitude	Dynamic	float32 array (16 values)	Band 2 memory effect magnitude measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_MAGNITUDES	B3_ME_Magnitude	Dynamic	float32 array (16 values)	Band 3 memory effect magnitude measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_MAGNITUDES	B4_ME_Magnitude	Dynamic	float32 array (16 values)	Band 3 memory effect magnitude measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_MAGNITUDES	B5_ME_Magnitude	Dynamic	float32 array (16 values)	Band 3 memory effect magnitude measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_MAGNITUDES	B6_ME_Magnitude	Dynamic	float32 array (8 values)	Band 3 memory effect magnitude measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_MAGNITUDES	B7_ME_Magnitude	Dynamic	float32 array (16 values)	Band 3 memory effect magnitude measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_MAGNITUDES	B8_ME_Magnitude	Dynamic	float32 array (32 values)	Band 3 memory effect magnitude measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: MEMORY_EFFECT GROUP: ME_TIME_CONSTANTS	B1_ME_Time_ Constant	Dynamic	float32 array (16 values)	Band 1 time constant measured in minor frames Valid format: NNNN.NNNNNNN, where NNNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_TIME_CONSTANTS	B2_ME_Time_ Constant	Dynamic	float32 array (16 values)	Band 2 time constant measured in minor frames Valid format: NNNN.NNNNNNN, where NNNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_TIME_CONSTANTS	B3_ME_Time_ Constant	Dynamic	float32 array (16 values)	Band 3 time constant measured in minor frames Valid format: NNNN.NNNNNNN, where NNNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_TIME_CONSTANTS	B4_ME_Time_ Constant	Dynamic	float32 array (16 values)	Band 4 time constant measured in minor frames Valid format: NNNN.NNNNNNN, where NNNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_TIME_CONSTANTS	B5_ME_Time_ Constant	Dynamic	float32 array (16 values)	Band 5 time constant measured in minor frames Valid format: NNNN.NNNNNNN, where NNNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_TIME_CONSTANTS	B6_ME_Time_ Constant	Dynamic	float32 array (8 values)	Band 6 time constant measured in minor frames Valid format: NNNN.NNNNNNN, where NNNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_TIME_CONSTANTS	B7_ME_Time_ Constant	Dynamic	float32 array (16 values)	Band 7 time constant measured in minor frames Valid format: NNNN.NNNNNNN, where NNNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: MEMORY_EFFECT GROUP: ME_TIME_CONSTANTS	B8_ME_Time_ Constant	Dynamic	float32 array (32 values)	Band 8 time constant measured in minor frames Valid format: NNNN.NNNNNNN, where NNNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: GHOST_PULSE	Ghost_Pulse_ Endpoints	Dynamic	float32 array (2 values)	Beginning and ending fractional minor frames that bound IC ghost pulse Valid format: NNNN.NNNN, where NNNN.NNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT	SCS_Reference_ Detectors	Dynamic	uint8 array (7 values)	Scan correlated shift reference detector, one per band Valid format: NN, where NN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_LOW	B1L_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 1 low-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_LOW	B2L_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 2 low-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_LOW	B3L_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 3 low-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_LOW	B4L_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 4 low-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_LOW	B5L_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 5 low-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_LOW	B7L_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 7 low-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_LOW	B8L_SCS_Magnitudes	Dynamic	float32 array (32 values)	Magnitude of band 8 low-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_HIGH	B1H_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 1 high-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_HIGH	B2H_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 2 high-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_HIGH	B3H_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 3 high-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_HIGH	B4H_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 4 high-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_HIGH	B5H_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 5 high-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_HIGH	B7H_SCS_Magnitudes	Dynamic	float32 array (16 values)	Magnitude of band 7 high-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: SCAN_CORRELATED_SHIFT GROUP: SCS_HIGH	B8H_SCS_Magnitudes	Dynamic	float32 array (32 values)	Magnitude of band 8 high-gain shift in digital numbers Valid format: SNNN.NNNNNNN, where S = "+" or "-" and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: STRIPING GROUP: STRIPING_FLAG_LOW	Correction_Reference_B1_Low	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 1, low gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_LOW	Correction_Reference_B2_Low	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 2, low gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_LOW	Correction_Reference_B3_Low	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 3, low gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_LOW	Correction_Reference_B4_Low	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 4, low gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_LOW	Correction_Reference_B5_Low	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 5, low gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_LOW	Correction_Reference_B6_Low	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 6, low gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: STRIPING GROUP: STRIPING_FLAG_LOW	Correction_Reference_B7_Low	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 7 , low gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_LOW	Correction_Reference_B8_Low	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 8, low gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_HIGH	Correction_Reference_B1_High	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 1, high gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_HIGH	Correction_Reference_B2_High	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 2, high gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_HIGH	Correction_Reference_B3_High	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 3, high gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_HIGH	Correction_Reference_B4_High	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 4, high gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_HIGH	Correction_Reference_B5_High	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 5, high gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_HIGH	Correction_Reference_B6_High	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 6, high gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: STRIPING GROUP: STRIPING_FLAG_HIGH	Correction_Reference_B7_High	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 7, high gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: STRIPING GROUP: STRIPING_FLAG_HIGH	Correction_Reference_B8_High	Static	uint8	Striping correction methodology flag, relative to band average or reference detector, band 8, high gain Valid format: N, where N = 0 (band average) or 1 (reference detector)	Yes	Yes	Postlaunch
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_LOW	Detector_Noise_Level_B1_Low	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 1, low gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_LOW	Detector_Noise_Level_B2_Low	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 2, low gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_LOW	Detector_Noise_Level_B3_Low	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 3, low gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_LOW	Detector_Noise_Level_B4_Low	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 4, low gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_LOW	Detector_Noise_Level_B5_Low	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 5, low gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_LOW	Detector_Noise_Level_B6_Low	Dynamic	float32 array (8 values)	Standard deviation of shutter region data for each detector of band 6, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_LOW	Detector_Noise_Level_B7_Low	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 7, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_LOW	Detector_Noise_Level_B8_Low	Dynamic	float32 array (32 values)	Standard deviation of shutter region data for each detector of band 8, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_HIGH	Detector_Noise_Level_B1_High	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 1, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_HIGH	Detector_Noise_Level_B2_High	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 2, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_HIGH	Detector_Noise_Level_B3_High	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 3, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_HIGH	Detector_Noise_Level_B4_High	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 4, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_HIGH	Detector_Noise_Level_B5_High	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 5, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_HIGH	Detector_Noise_Level_B6_High	Dynamic	float32 array (8 values)	Standard deviation of shutter region data for each detector of band 6, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_HIGH	Detector_Noise_Level_B7_High	Dynamic	float32 array (16 values)	Standard deviation of shutter region data for each detector of band 7, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: DETECTOR_NOISE GROUP: DETECTOR_NOISE_HIGH	Detector_Noise_Level_B8_High	Dynamic	float32 array (32 values)	Standard deviation of shutter region data for each detector of band 8, high gain Valid format: NN.NNNN, where NN.NNNN = TBS	No	Yes	IAS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: HISTOGRAM GROUP: REFERENCE_DETECTORS	Reference_Detector_B1	Dynamic	uint8	Detector used as a reference when computing relative detector gains and biases (least noisy), band 1 Valid format: NN, where NN = TBS	Yes	Yes	IAS
GROUP: HISTOGRAM GROUP: REFERENCE_DETECTORS	Reference_Detector_B2	Dynamic	uint8	Detector used as a reference when computing relative detector gains and biases (least noisy), band 2 Valid format: NN, where NN = TBS	Yes	Yes	IAS
GROUP: HISTOGRAM GROUP: REFERENCE_DETECTORS	Reference_Detector_B3	Dynamic	uint8	Detector used as a reference when computing relative detector gains and biases (least noisy), band 3 Valid format: NN, where NN = TBS	Yes	Yes	IAS
GROUP: HISTOGRAM GROUP: REFERENCE_DETECTORS	Reference_Detector_B4	Dynamic	uint8	Detector used as a reference when computing relative detector gains and biases (least noisy), band 4 Valid format: NN, where NN = TBS	Yes	Yes	IAS
GROUP: HISTOGRAM GROUP: REFERENCE_DETECTORS	Reference_Detector_B5	Dynamic	uint8	Detector used as a reference when computing relative detector gains and biases (least noisy), band 5 Valid format: NN, where NN = TBS	Yes	Yes	IAS
GROUP: HISTOGRAM GROUP: REFERENCE_DETECTORS	Reference_Detector_B6	Dynamic	uint8	Detector used as a reference when computing relative detector gains and biases (least noisy), band 6 Valid format: NN, where NN = TBS	Yes	Yes	IAS
GROUP: HISTOGRAM GROUP: REFERENCE_DETECTORS	Reference_Detector_B7	Dynamic	uint8	Detector used as a reference when computing relative detector gains and biases (least noisy), band 7 Valid format: NN, where NN = TBS	Yes	Yes	IAS
GROUP: HISTOGRAM GROUP: REFERENCE_DETECTORS	Reference_Detector_B8	Dynamic	uint8	Detector used as a reference when computing relative detector gains and biases (least noisy), band 8 Valid format: NN, where NN = TBS	Yes	Yes	IAS
GROUP: HISTOGRAM GROUP: SATURATION_THRESHOLDS	Saturation_Bin_Threshold_B1	Dynamic	uint16	Number of pixels that a bin must have to be tested as a saturation bin, band 1 Valid format: NNNNN, where NNNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: SATURATION_THRESHOLDS	Saturation_Bin_Threshold_B2	Dynamic	uint16	Number of pixels that a bin must have to be tested as a saturation bin, band 2 Valid format: NNNNN, where NNNNN = TBS	No	Yes	IAS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: HISTOGRAM GROUP: SATURATION_THRESHOLDS	Saturation_Bin_Threshold_B3	Dynamic	uint16	Number of pixels that a bin must have to be tested as a saturation bin, band 3 Valid format: NNNNN, where NNNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: SATURATION_THRESHOLDS	Saturation_Bin_Threshold_B4	Dynamic	uint16	Number of pixels that a bin must have to be tested as a saturation bin, band 4 Valid format: NNNNN, where NNNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: SATURATION_THRESHOLDS	Saturation_Bin_Threshold_B5	Dynamic	uint16	Number of pixels that a bin must have to be tested as a saturation bin, band 5 Valid format: NNNNN, where NNNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: SATURATION_THRESHOLDS	Saturation_Bin_Threshold_B6	Dynamic	uint16	Number of pixels that a bin must have to be tested as a saturation bin, band 6 Valid format: NNNNN, where NNNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: SATURATION_THRESHOLDS	Saturation_Bin_Threshold_B7	Dynamic	uint16	Number of pixels that a bin must have to be tested as a saturation bin, band 7 Valid format: NNNNN, where NNNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: SATURATION_THRESHOLDS	Saturation_Bin_Threshold_B8	Dynamic	uint16	Number of pixels that a bin must have to be tested as a saturation bin, band 8 Valid format: NN, where NNNN = TBS	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_NUMBER	Adjacent_Bin_Number_B1	Dynamic	uint8	Bins adjacent to possible saturation bin that must have fewer pixels than "adjacent bin threshold" to declare possible bin as saturation bin, band 1 Valid format: N, where N = 2(default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_NUMBER	Adjacent_Bin_Number_B2	Dynamic	uint8	Bins adjacent to possible saturation bin that must have fewer pixels than "adjacent bin threshold" to declare possible bin as saturation bin, band 2 Valid format: N, where N = 2(default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_NUMBER	Adjacent_Bin_Number_B3	Dynamic	uint8	Bins adjacent to possible saturation bin that must have fewer pixels than "adjacent bin threshold" to declare possible bin as saturation bin, band 3 Valid format: N, where N = 2(default)	No	Yes	IAS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_NUMBER	Adjacent_Bin_ Number_B4	Dynamic	uint8	Bins adjacent to possible saturation bin that must have fewer pixels than "adjacent bin threshold" to declare possible bin as saturation bin, band 4 Valid format: N, where N = 2(default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_NUMBER	Adjacent_Bin_ Number_B5	Dynamic	uint8	Bins adjacent to possible saturation bin that must have fewer pixels than "adjacent bin threshold" to declare possible bin as saturation bin, band 5 Valid format: N, where N = 2(default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_NUMBER	Adjacent_Bin_ Number_B6	Dynamic	uint8	Bins adjacent to possible saturation bin that must have fewer pixels than "adjacent bin threshold" to declare possible bin as saturation bin, band 6 Valid format: N, where N = 2(default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_NUMBER	Adjacent_Bin_ Number_B7	Dynamic	uint8	Bins adjacent to possible saturation bin that must have fewer pixels than "adjacent bin threshold" to declare possible bin as saturation bin, band 7 Valid format: N, where N = 2(default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_NUMBER	Adjacent_Bin_ Number_B8	Dynamic	uint8	Bins adjacent to possible saturation bin that must have fewer pixels than "adjacent bin threshold" to declare possible bin as saturation bin, band 8 Valid format: N, where N = 2(default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_THRESHOLD	Adjacent_Bin_ Threshold_B1	Dynamic	uint8	Number of adjacent bin pixels that cannot be exceeded for band 1 candidate saturation bin to be valid saturation bin Valid format: NN, where NN = 10 (default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_THRESHOLD	Adjacent_Bin_ Threshold_B2	Dynamic	uint8	Number of adjacent bin pixels that cannot be exceeded for band 2 candidate saturation bin to be valid saturation bin Valid format: NN, where NN = 10 (default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_THRESHOLD	Adjacent_Bin_ Threshold_B3	Dynamic	uint8	Number of adjacent bin pixels that cannot be exceeded for band 3 candidate saturation bin to be valid saturation bin Valid format: NN, where NN = 10 (default)	No	Yes	IAS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_THRESHOLD	Adjacent_Bin_ Threshold_B4	Dynamic	uint8	Number of adjacent bin pixels that cannot be exceeded for band 4 candidate saturation bin to be valid saturation bin Valid format: NN, where NN = 10 (default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_THRESHOLD	Adjacent_Bin_ Threshold_B5	Dynamic	uint8	Number of adjacent bin pixels that cannot be exceeded for band 5 candidate saturation bin to be valid saturation bin Valid format: NN, where NN = 10 (default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_THRESHOLD	Adjacent_Bin_ Threshold_B6	Dynamic	uint8	Number of adjacent bin pixels that cannot be exceeded for band 6 candidate saturation bin to be valid saturation bin Valid format: NN, where NN = 10 (default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_THRESHOLD	Adjacent_Bin_ Threshold_B7	Dynamic	uint8	Number of adjacent bin pixels that cannot be exceeded for band 7 candidate saturation bin to be valid saturation bin Valid format: NN, where NN = 10 (default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: ADJACENT_BINS GROUP: BIN_THRESHOLD	Adjacent_Bin_ Threshold_B8	Dynamic	uint8	Number of adjacent bin pixels that cannot be exceeded for band 8 candidate saturation bin to be valid saturation bin Valid format: NN, where NN = 10 (default)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: STARTING_PIXEL	Start_pixel_B1	Dynamic	uint8	Leftmost pixel in window to be tested, band 1 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: STARTING_PIXEL	Start_pixel_B2	Dynamic	uint8	Leftmost pixel in window to be tested, band 2 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: STARTING_PIXEL	Start_pixel_B3	Dynamic	uint8	Leftmost pixel in window to be tested, band 3 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: STARTING_PIXEL	Start_pixel_B4	Dynamic	uint8	Leftmost pixel in window to be tested, band 4 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: STARTING_PIXEL	Start_pixel_B5	Dynamic	uint8	Leftmost pixel in window to be tested, band 5 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: STARTING_PIXEL	Start_pixel_B6	Dynamic	uint8	Leftmost pixel in window to be tested, band 6 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: HISTOGRAM GROUP: STARTING_PIXEL	Start_pixel_B7	Dynamic	uint8	Leftmost pixel in window to be tested, band 7 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: STARTING_PIXEL	Start_pixel_B8	Dynamic	uint8	Leftmost pixel in window to be tested, band 8 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_WIDTH	Window_Samples_B1	Dynamic	uint8	Width of window, in pixels, to be tested, band1 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_WIDTH	Window_Samples_B2	Dynamic	uint8	Width of window, in pixels, to be tested, band2 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_WIDTH	Window_Samples_B3	Dynamic	uint8	Width of window, in pixels, to be tested, band3 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_WIDTH	Window_Samples_B4	Dynamic	uint8	Width of window, in pixels, to be tested, band4 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_WIDTH	Window_Samples_B5	Dynamic	uint8	Width of window, in pixels, to be tested, band5 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_WIDTH	Window_Samples_B6	Dynamic	uint8	Width of window, in pixels, to be tested, band6 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_WIDTH	Window_Samples_B7	Dynamic	uint8	Width of window, in pixels, to be tested, band7 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_WIDTH	Window_Samples_B8	Dynamic	uint8	Width of window, in pixels, to be tested, band8 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_LENGTH	Window_Scans_B1	Dynamic	uint8	Number of scans in window to be tested, band1 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_LENGTH	Window_Scans_B2	Dynamic	uint8	Number of scans in window to be tested, band2 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_LENGTH	Window_Scans_B3	Dynamic	uint8	Number of scans in window to be tested, band3 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_LENGTH	Window_Scans_B4	Dynamic	uint8	Number of scans in window to be tested, band4 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: HISTOGRAM GROUP: WINDOW_LENGTH	Window_Scans_B5	Dynamic	uint8	Number of scans in window to be tested, band5 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_LENGTH	Window_Scans_B6	Dynamic	uint8	Number of scans in window to be tested, band6 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_LENGTH	Window_Scans_B7	Dynamic	uint8	Number of scans in window to be tested, band7 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: WINDOW_LENGTH	Window_Scans_B8	Dynamic	uint8	Number of scans in window to be tested, band8 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: OVERLAPPING_SCANS	Overlap_Scans_B1	Dynamic	uint8	Number of overlapping scans between windows to be tested, band 1 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: OVERLAPPING_SCANS	Overlap_Scans_B2	Dynamic	uint8	Number of overlapping scans between windows to be tested, band 2 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: OVERLAPPING_SCANS	Overlap_Scans_B3	Dynamic	uint8	Number of overlapping scans between windows to be tested, band 3 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: OVERLAPPING_SCANS	Overlap_Scans_B4	Dynamic	uint8	Number of overlapping scans between windows to be tested, band 4 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: OVERLAPPING_SCANS	Overlap_Scans_B5	Dynamic	uint8	Number of overlapping scans between windows to be tested, band 5 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: OVERLAPPING_SCANS	Overlap_Scans_B6	Dynamic	uint8	Number of overlapping scans between windows to be tested, band 6 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: OVERLAPPING_SCANS	Overlap_Scans_B7	Dynamic	uint8	Number of overlapping scans between windows to be tested, band 7 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: HISTOGRAM GROUP: OVERLAPPING_SCANS	Overlap_Scans_B8	Dynamic	uint8	Number of overlapping scans between windows to be tested, band 8 Valid format: NNN, where NNN = (TBS)	No	Yes	IAS
GROUP: IMPULSE_NOISE	Median_Filter_Width	Static	uint8	Width of median filter Valid format: N, where N = 3	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B1L_Threshold	Dynamic	float32 array (16 values)	Band 1 low-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B2L_Threshold	Dynamic	float32 array (16 values)	Band 2 low-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B3L_Threshold	Dynamic	float32 array (16 values)	Band 3 low-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B4L_Threshold	Dynamic	float32 array (16 values)	Band 4 low-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B5L_Threshold	Dynamic	float32 array (16 values)	Band 5 low-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B6L_Threshold	Dynamic	float32 array (8 values)	Band 6 low-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B7L_Threshold	Dynamic	float32 array (16 values)	Band 7 low-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B8L_Threshold	Dynamic	float32 array (32 values)	Band 8 low-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B1H_Threshold	Dynamic	float32 array (16 values)	Band 1 high-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B2H_Threshold	Dynamic	float32 array (16 values)	Band 2 high-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B3H_Threshold	Dynamic	float32 array (16 values)	Band 3 high-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B4H_Threshold	Dynamic	float32 array (16 values)	Band 4 high-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B5H_Threshold	Dynamic	float32 array (16 values)	Band 5 high-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B6H_Threshold	Dynamic	float32 array (8 values)	Band 6 high-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B7H_Threshold	Dynamic	float32 array (16 values)	Band 7 high-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: IMPULSE_NOISE GROUP: IN_THRESHOLD	B8H_Threshold	Dynamic	float32 array (32 values)	Band 8 high-gain noise threshold Valid format: NN.NNNNNNN, where NN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE	Frequency_Components	Dynamic	uint8	Number of frequency components derived during waveform analysis for coherent noise correction Valid format: NN, where NN = 10	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MEANS	B1_Frequency_Mean	Dynamic	float32 array (10 values)	Band 1 frequency means measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MEANS	B2_Frequency_Mean	Dynamic	float32 array (10 values)	Band 2 frequency means measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MEANS	B3_Frequency_Mean	Dynamic	float32 array (10 values)	Band 3 frequency means measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MEANS	B4_Frequency_Mean	Dynamic	float32 array (10 values)	Band 4 frequency means measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MEANS	B5_Frequency_Mean	Dynamic	float32 array (10 values)	Band 5 frequency means measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MEANS	B6_Frequency_Mean	Dynamic	float32 array (10 values)	Band 6 frequency means measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MEANS	B7_Frequency_Mean	Dynamic	float32 array (10 values)	Band 7 frequency means measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MEANS	B8_Frequency_Mean	Dynamic	float32 array (10 values)	Band 8 frequency means measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_SIGMAS	B1_Frequency_Sigma	Dynamic	float32 array (10 values)	Band 1 frequency sigmas measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_SIGMAS	B2_Frequency_Sigma	Dynamic	float32 array (10 values)	Band 2 frequency sigmas measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_SIGMAS	B3_Frequency_Sigma	Dynamic	float32 array (10 values)	Band 3 frequency sigmas measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_SIGMAS	B4_Frequency_Sigma	Dynamic	float32 array (10 values)	Band 4 frequency sigmas measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_SIGMAS	B5_Frequency_Sigma	Dynamic	float32 array (10 values)	Band 5 frequency sigmas measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_SIGMAS	B6_Frequency_Sigma	Dynamic	float32 array (10 values)	Band 6 frequency sigmas measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_SIGMAS	B7_Frequency_Sigma	Dynamic	float32 array (10 values)	Band 7 frequency sigmas measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_SIGMAS	B8_Frequency_Sigma	Dynamic	float32 array (10 values)	Band 8 frequency sigmas measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MINIMUMS	B1_Frequency_Min	Dynamic	float32 array (10 values)	Band 1 frequency minimums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MINIMUMS	B2_Frequency_Min	Dynamic	float32 array (10 values)	Band 2 frequency minimums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MINIMUMS	B3_Frequency_Min	Dynamic	float32 array (10 values)	Band 3 frequency minimums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MINIMUMS	B4_Frequency_Min	Dynamic	float32 array (10 values)	Band 4 frequency minimums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MINIMUM S	B5_Frequency_Min	Dynamic	float32 array (10 values)	Band 5 frequency minimums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MINIMUM S	B6_Frequency_Min	Dynamic	float32 array (10 values)	Band 6 frequency minimums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MINIMUM S	B7_Frequency_Min	Dynamic	float32 array (10 values)	Band 7 frequency minimums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MINIMUM S	B8_Frequency_Min	Dynamic	float32 array (10 values)	Band 8 frequency minimums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MAXIMUM S	B1_Frequency_Max	Dynamic	float32 array (10 values)	Band 1 frequency maximums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MAXIMUM S	B2_Frequency_Max	Dynamic	float32 array (10 values)	Band 2 frequency maximums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MAXIMUM S	B3_Frequency_Max	Dynamic	float32 array (10 values)	Band 3 frequency maximums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MAXIMUM S	B4_Frequency_Max	Dynamic	float32 array (10 values)	Band 4 frequency maximums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MAXIMUM S	B5_Frequency_Max	Dynamic	float32 array (10 values)	Band 5 frequency maximums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MAXIMUM S	B6_Frequency_Max	Dynamic	float32 array (10 values)	Band 6 frequency maximums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MAXIMUM S	B7_Frequency_Max	Dynamic	float32 array (10 values)	Band 7 frequency maximums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_FREQUENCY_ PARAMETERS GROUP: FREQUENCY_MAXIMUM S	B8_Frequency_Max	Dynamic	float32 array (10 values)	Band 8 frequency maximums measured in inverse minor frames Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETE RS GROUP: PHASE_MEANS	B1_Phase_Mean	Dynamic	float32 array (10 values)	Band 1 phase means measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETE RS GROUP: PHASE_MEANS	B2_Phase_Mean	Dynamic	float32 array (10 values)	Band 2 phase means measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETE RS GROUP: PHASE_MEANS	B3_Phase_Mean	Dynamic	float32 array (10 values)	Band 3 phase means measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MEANS	B4_Phase_Mean	Dynamic	float32 array (10 values)	Band 4 phase means measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MEANS	B5_Phase_Mean	Dynamic	float32 array (10 values)	Band 5 phase means measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MEANS	B6_Phase_Mean	Dynamic	float32 array (10 values)	Band 6 phase means measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MEANS	B7_Phase_Mean	Dynamic	float32 array (10 values)	Band 7 phase means measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MEANS	B8_Phase_Mean	Dynamic	float32 array (10 values)	Band 8 phase means measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_SIGMAS	B1_Phase_Sigma	Dynamic	float32 array (10 values)	Band 1 phase sigmas measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_SIGMAS	B2_Phase_Sigma	Dynamic	float32 array (10 values)	Band 2 phase sigmas measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_SIGMAS	B3_Phase_Sigma	Dynamic	float32 array (10 values)	Band 3 phase sigmas measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_SIGMAS	B4_Phase_Sigma	Dynamic	float32 array (10 values)	Band 4 phase sigmas measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_SIGMAS	B5_Phase_Sigma	Dynamic	float32 array (10 values)	Band 5 phase sigmas measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_SIGMAS	B6_Phase_Sigma	Dynamic	float32 array (10 values)	Band 6 phase sigmas measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_SIGMAS	B7_Phase_Sigma	Dynamic	float32 array (10 values)	Band 7 phase sigmas measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_SIGMAS	B8_Phase_Sigma	Dynamic	float32 array (10 values)	Band 8 phase sigmas measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MINIMUMS	B1_Phase_Min	Dynamic	float32 array (10 values)	Band 1 phase minimums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MINIMUMS	B2_Phase_Min	Dynamic	float32 array (10 values)	Band 2 phase minimums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MINIMUMS	B3_Phase_Min	Dynamic	float32 array (10 values)	Band 3 phase minimums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MINIMUMS	B4_Phase_Min	Dynamic	float32 array (10 values)	Band 4 phase minimums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MINIMUMS	B5_Phase_Min	Dynamic	float32 array (10 values)	Band 5 phase minimums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MINIMUMS	B6_Phase_Min	Dynamic	float32 array (10 values)	Band 6 phase minimums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MINIMUMS	B7_Phase_Min	Dynamic	float32 array (10 values)	Band 7 phase minimums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MINIMUMS	B8_Phase_Min	Dynamic	float32 array (10 values)	Band 8 phase minimums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MAXIMUMS	B1_Phase_Max	Dynamic	float32 array (10 values)	Band 1 phase maximums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MAXIMUMS	B2_Phase_Max	Dynamic	float32 array (10 values)	Band 2 phase maximums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MAXIMUMS	B3_Phase_Max	Dynamic	float32 array (10 values)	Band 3 phase maximums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MAXIMUMS	B4_Phase_Max	Dynamic	float32 array (10 values)	Band 4 phase maximums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MAXIMUMS	B5_Phase_Max	Dynamic	float32 array (10 values)	Band 5 phase maximums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MAXIMUMS	B6_Phase_Max	Dynamic	float32 array (10 values)	Band 6 phase maximums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MAXIMUMS	B7_Phase_Max	Dynamic	float32 array (10 values)	Band 7 phase maximums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_PHASE_PARAMETERS GROUP: PHASE_MAXIMUMS	B8_Phase_Max	Dynamic	float32 array (10 values)	Band 8 phase maximums measured in radians Valid format: NNNNNNN, where NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_PARAMETERS GROUP: MAGNITUDE_MEANS	B1_Magnitude_Mean	Dynamic	float32 array (10 values)	Band 1 magnitudes means measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_PARAMETERS GROUP: MAGNITUDE_MEANS	B2_Magnitude_Mean	Dynamic	float32 array (10 values)	Band 2 magnitudes means measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_PARAMETERS GROUP: MAGNITUDE_MEANS	B3_Magnitude_Mean	Dynamic	float32 array (10 values)	Band 3 magnitudes means measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_PARAMETERS GROUP: MAGNITUDE_MEANS	B4_Magnitude_Mean	Dynamic	float32 array (10 values)	Band 4 magnitudes means measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MEANS	B5_Magnitude_Mean	Dynamic	float32 array (10 values)	Band 5 magnitudes means measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MEANS	B6_Magnitude_Mean	Dynamic	float32 array (10 values)	Band 6 magnitudes means measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MEANS	B7_Magnitude_Mean	Dynamic	float32 array (10 values)	Band 7 magnitudes means measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MEANS	B8_Magnitude_Mean	Dynamic	float32 array (10 values)	Band 8 magnitudes means measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_SIGMAS	B1_Magnitude_Sigma	Dynamic	float32 array (10 values)	Band 1 magnitudes sigmas measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_SIGMAS	B2_Magnitude_Sigma	Dynamic	float32 array (10 values)	Band 2 magnitudes sigmas measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_SIGMAS	B3_Magnitude_Sigma	Dynamic	float32 array (10 values)	Band 3 magnitudes sigmas measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_SIGMAS	B4_Magnitude_Sigma	Dynamic	float32 array (10 values)	Band 4 magnitudes sigmas measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_SIGMAS	B5_Magnitude_Sigma	Dynamic	float32 array (10 values)	Band 5 magnitudes sigmas measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_SIGMAS	B6_Magnitude_Sigma	Dynamic	float32 array (10 values)	Band 6 magnitudes sigmas measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_SIGMAS	B7_Magnitude_Sigma	Dynamic	float32 array (10 values)	Band 7 magnitudes sigmas measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_SIGMAS	B8_Magnitude_Sigma	Dynamic	float32 array (10 values)	Band 8 magnitudes sigmas measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MINIMUMS	B1_Magnitude_Min	Dynamic	float32 array (10 values)	Band 1 magnitudes minimums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MINIMUMS	B2_Magnitude_Min	Dynamic	float32 array (10 values)	Band 2 magnitudes minimums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MINIMUMS	B3_Magnitude_Min	Dynamic	float32 array (10 values)	Band 3 magnitudes minimums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MINIMUMS	B4_Magnitude_Min	Dynamic	float32 array (10 values)	Band 4 magnitudes minimums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MINIMUMS	B5_Magnitude_Min	Dynamic	float32 array (10 values)	Band 5 magnitudes minimums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MINIMUMS	B6_Magnitude_Min	Dynamic	float32 array (10 values)	Band 6 magnitudes minimums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MINIMUMS	B7_Magnitude_Min	Dynamic	float32 array (10 values)	Band 7 magnitudes minimums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MINIMUMS	B8_Magnitude_Min	Dynamic	float32 array (10 values)	Band 8 magnitudes minimums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MAXIMUM S	B1_Magnitude_Max	Dynamic	float32 array (10 values)	Band 1 magnitudes maximums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MAXIMUM S	B2_Magnitude_Max	Dynamic	float32 array (10 values)	Band 2 magnitudes maximums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MAXIMUM S	B3_Magnitude_Max	Dynamic	float32 array (10 values)	Band 3 magnitudes maximums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MAXIMUM S	B4_Magnitude_Max	Dynamic	float32 array (10 values)	Band 4 magnitudes maximums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MAXIMUM S	B5_Magnitude_Max	Dynamic	float32 array (10 values)	Band 5 magnitudes maximums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MAXIMUM S	B6_Magnitude_Max	Dynamic	float32 array (10 values)	Band 6 magnitudes maximums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MAXIMUM S	B7_Magnitude_Max	Dynamic	float32 array (10 values)	Band 7 magnitudes maximums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: COHERENT_NOISE GROUP: CN_MAGNITUDE_ PARAMETERS GROUP: MAGNITUDE_MAXIMUM S	B8_Magnitude_Max	Dynamic	float32 array (10 values)	Band 8 magnitudes maximums measured in DNs Valid format: NNN.NNNNNNN, where NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: DETECTOR_SATURATIO N GROUP: AD_CONVERTER_SATU RATION GROUP: AD_CONVERTER_ SATURATION_LOW	High_AD_Level_B1_ low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 1, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATIO N GROUP: AD_CONVERTER_SATU RATION GROUP: AD_CONVERTER_ SATURATION_LOW	High_AD_Level_B2_ low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 2, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATIO N GROUP: AD_CONVERTER_SATU RATION GROUP: AD_CONVERTER_ SATURATION_LOW	High_AD_Level_B3_ low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 3, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	High_AD_Level_B4_low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 4, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	High_AD_Level_B5_low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 5, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	High_AD_Level_B6_low	Dynamic	uint8 array (8 values)	Digital count at which analog-to-digital converter saturates at high end; band 6, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	High_AD_Level_B7_low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 7, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	High_AD_Level_B8_low	Dynamic	uint8 array (32 values)	Digital count at which analog-to-digital converter saturates at high end; band 8, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	Low_AD_Level_B1_low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 1, low gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	Low_AD_Level_B2_low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 2, low gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	Low_AD_Level_B3_low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 3, low gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	Low_AD_Level_B4_low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 4, low gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	Low_AD_Level_B5_low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 5, low gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	Low_AD_Level_B6_low	Dynamic	uint8 array (8 values)	Digital count at which analog-to-digital converter saturates at low end; band 6, low gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	Low_AD_Level_B7_low	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 7, low gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_LOW	Low_AD_Level_B8_low	Dynamic	uint8 array (32 values)	Digital count at which analog-to-digital converter saturates at low end; band 8, low gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	High_AD_Level_B1_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 1, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	High_AD_Level_B2_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 2, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	High_AD_Level_B3_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 3, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	High_AD_Level_B4_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 4, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	High_AD_Level_B5_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 5, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	High_AD_Level_B6_high	Dynamic	uint8 array (8 values)	Digital count at which analog-to-digital converter saturates at high end; band 6, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	High_AD_Level_B7_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at high end; band 7, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	High_AD_Level_B8_high	Dynamic	uint8 array (32 values)	Digital count at which analog-to-digital converter saturates at high end; band 8, gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	Low_AD_Level_B1_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 1, high gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	Low_AD_Level_B2_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 2, high gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	Low_AD_Level_B3_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 3, high gain Valid format: NNN, where NNN = 000 (default	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	Low_AD_Level_B4_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 4, high gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	Low_AD_Level_B5_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 5, high gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	Low_AD_Level_B6_high	Dynamic	uint8 array (8 values)	Digital count at which analog-to-digital converter saturates at low end; band 6, high gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	Low_AD_Level_B7_high	Dynamic	uint8 array (16 values)	Digital count at which analog-to-digital converter saturates at low end; band 7, high gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: AD_CONVERTER_SATURATION GROUP: AD_CONVERTER_SATURATION_HIGH	Low_AD_Level_B8_high	Dynamic	uint8 array (32 values)	Digital count at which analog-to-digital converter saturates at low end; band 8, gain Valid format: NNN, where NNN = 000 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	High_Analog_Level_B1_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 1, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	High_Analog_Level_B2_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 2, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	High_Analog_Level_B3_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 3, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	High_Analog_Level_B4_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 4, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	High_Analog_Level_B5_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 5, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	High_Analog_Level_B6_low	Dynamic	uint8 array (8 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 6, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	High_Analog_Level_B7_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 7, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	High_Analog_Level_B8_low	Dynamic	uint8 array (32 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 8, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	Low_Analog_Level_B1_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 1, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	Low_Analog_Level_B2_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 2, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	Low_Analog_Level_B3_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 3, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	Low_Analog_Level_B4_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 4, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	Low_Analog_Level_B5_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 5, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	Low_Analog_Level_B6_low	Dynamic	uint8 array (8 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 6, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	Low_Analog_Level_B7_low	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 7, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_LOW	Low_Analog_Level_B8_low	Dynamic	uint8 array (32 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 8, low gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	High_Analog_Level_B1_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 1, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	High_Analog_Level_B2_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 2, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	High_Analog_Level_B3_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 3, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	High_Analog_Level_B4_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 4, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	High_Analog_Level_B5_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 5, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	High_Analog_Level_B6_high	Dynamic	uint8 array (8 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 6, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	High_Analog_Level_B7_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 7, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	High_Analog_Level_B8_high	Dynamic	uint8 array (32 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at high end; band 8, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	Low_Analog_Level_B1_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 1, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	Low_Analog_Level_B2_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 2, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	Low_Analog_Level_B3_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 3, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	Low_Analog_Level_B4_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 4, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	Low_Analog_Level_B5_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 5, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	Low_Analog_Level_B6_high	Dynamic	uint8 array (8 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 6, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	Low_Analog_Level_B7_high	Dynamic	uint8 array (16 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 7, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: DETECTOR_SATURATION GROUP: ANALOG_SIGNAL_SATURATION GROUP: ANALOG_SIGNAL_SATURATION_HIGH	Low_Analog_Level_B8_high	Dynamic	uint8 array (32 values)	Digital count corresponding to signal level at which analog portion of signal chain saturates at low end; band 8, high gain Valid format: NNN, where NNN = 255 (default)	Yes	Yes	SBRS
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B1L_RTemp_Prelaunch	Static	float64	Band 1 prelaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B1L_RTemp_Postlaunch	Static	float64	Band 1 postlaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B1L_RTemp_Current	Dynamic	float64	Band 1 current low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B2L_RTemp_Prelaunch	Static	float64	Band 2 prelaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B2L_RTemp_Postlaunch	Static	float64	Band 2 postlaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B2L_RTemp_Current	Dynamic	float64	Band 2 current low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B3L_RTemp_Prelaunch	Static	float64	Band 3 prelaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B3L_RTemp_Postlaunch	Static	float64	Band 3 postlaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B3L_RTemp_Current	Dynamic	float64	Band 3 current low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B4L_RTemp_ Prelaunch	Static	float64	Band 4 prelaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B4L_RTemp_ Postlaunch	Static	float64	Band 4 postlaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B4L_RTemp_Current	Dynamic	float64	Band 4 current low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B5L_RTemp_ Prelaunch	Static	float64	Band 5 prelaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B5L_RTemp_ Postlaunch	Static	float64	Band 5 postlaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B5L_RTemp_Current	Dynamic	float64	Band 5 current low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B6L_RTemp_ Prelaunch	Static	float64	Band 6 prelaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	BL10 in thermal vacuum
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B6L_RTemp_ Postlaunch	Static	float64	Band 6 postlaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B6L_RTemp_Current	Dynamic	float64	Band 6 current low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	BL10 in thermal vacuum

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B7L_RTemp_ Prelaunch	Static	float64	Band 7 prelaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B7L_RTemp_ Postlaunch	Static	float64	Band 7 postlaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B7L_RTemp_Current	Dynamic	float64	Band 7 current low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B8L_RTemp_ Prelaunch	Static	float64	Band 8 prelaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B8L_RTemp_ Postlaunch	Static	float64	Band 8 postlaunch low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_LOW	B8L_RTemp_Current	Dynamic	float64	Band 8 current low-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B1H_RTemp_ Prelaunch	Static	float64	Band 1 prelaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B1H_RTemp_ Postlaunch	Static	float64	Band 1 postlaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B1H_RTemp_Current	Dynamic	float64	Band 1 current high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B2H_RTemp_ Prelaunch	Static	float64	Band 2 prelaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B2H_RTemp_ Postlaunch	Static	float64	Band 2 postlaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B2H_RTemp_Current	Dynamic	float64	Band 2 current high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B3H_RTemp_ Prelaunch	Static	float64	Band 3 prelaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B3H_RTemp_ Postlaunch	Static	float64	Band 3 postlaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B3H_RTemp_Current	Dynamic	float64	Band 3 current high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B4H_RTemp_ Prelaunch	Static	float64	Band 4 prelaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B4H_RTemp_ Postlaunch	Static	float64	Band 4 postlaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B4H_RTemp_Current	Dynamic	float64	Band 4 current high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B5H_RTemp_ Prelaunch	Static	float64	Band 5 prelaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B5H_RTemp_ Postlaunch	Static	float64	Band 5 postlaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B5H_RTemp_Current	Dynamic	float64	Band 5 current high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B6H_RTemp_ Prelaunch	Static	float64	Band 6 prelaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B6H_RTemp_ Postlaunch	Static	float64	Band 6 postlaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B6H_RTemp_Current	Dynamic	float64	Band 6 current high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B7H_RTemp_ Prelaunch	Static	float64	Band 7 prelaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B7H_RTemp_ Postlaunch	Static	float64	Band 7 postlaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B7H_RTemp_Current	Dynamic	float64	Band 7 current high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B8H_RTemp_ Prelaunch	Static	float64	Band 8 prelaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B8H_RTemp_ Postlaunch	Static	float64	Band 8 postlaunch high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: REFERENCE_TEMPERATURES GROUP: REFERENCE_HIGH	B8H_RTemp_Current	Dynamic	float64	Band 8 current high-gain calibration reference temperature in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS	No	Yes	AC02 telemetry

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B1L_SCoeff_Prelaunch	Static	float64 array (16 values)	Band 1 prelaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B1L_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 1 postlaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B1L_SCoeff_Current	Dynamic	float64 array (16 values)	Band 1 current low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B2L_SCoeff_Prelaunch	Static	float64 array (16 values)	Band 2 prelaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B2L_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 2 postlaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B2L_SCoeff_Current	Dynamic	float64 array (16 values)	Band 2 current low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B3L_SCoeff_Prelaunch	Static	float64 array (16 values)	Band 3 prelaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B3L_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 3 postlaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B3L_SCoeff_Current	Dynamic	float64 array (16 values)	Band 3 current low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B4L_SCoeff_Prelaunch	Static	float64 array (16 values)	Band 4 prelaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B4L_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 4 postlaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B4L_SCoeff_Current	Dynamic	float64 array (16 values)	Band 4 current low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B5L_SCoeff_Prelaunch	Static	float64 array (16 values)	Band 5 prelaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B5L_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 5 postlaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B5L_SCoeff_Current	Dynamic	float64 array (16 values)	Band 5 current low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B6L_SCoeff_Prelaunch	Static	float64 array (8 values)	Band 6 prelaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B6L_SCoeff_ Postlaunch	Static	float64 array (8 values)	Band 6 postlaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B6L_SCoeff_Current	Dynamic	float64 array (8 values)	Band 6 current low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B6L_SCoeffOff_ Prelaunch	Static	float64 array (8 values)	Band 6 prelaunch offset calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B6L_SCoeffOff_ Postlaunch	Static	float64 array (8 values)	Band 6 postlaunch offset calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B6L_SCoeffOff_ Current	Dynamic	float64 array (8 values)	Band 6 current offset calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B7L_SCoeff_Prelaunch	Static	float64 array (16 values)	Band 7 prelaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B7L_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 7 postlaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B7L_SCoeff_Current	Dynamic	float64 array (16 values)	Band 7 current low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B8L_SCoeff_Prelaunch	Static	float64 array (32 values)	Band 8 prelaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B8L_SCoeff_ Postlaunch	Static	float64 array (32 values)	Band 8 postlaunch low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_LOW	B8L_SCoeff_Current	Dynamic	float64 array (32 values)	Band 8 current low-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B1H_SCoeff_ Prelaunch	Static	float64 array (16 values)	Band 1 prelaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B1H_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 1 postlaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B1H_SCoeff_Current	Dynamic	float64 array (16 values)	Band 1 current high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B2H_SCoeff_ Prelaunch	Static	float64 array (16 values)	Band 2 prelaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B2H_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 2 postlaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B2H_SCoeff_Current	Dynamic	float64 array (16 values)	Band 2 current high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B3H_SCoeff_ Prelaunch	Static	float64 array (16 values)	Band 3 prelaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B3H_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 3 postlaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B3H_SCoeff_Current	Dynamic	float64 array (16 values)	Band 3 current high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B4H_SCoeff_ Prelaunch	Static	float64 array (16 values)	Band 4 prelaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B4H_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 4 postlaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B4H_SCoeff_Current	Dynamic	float64 array (16 values)	Band 4 current high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B5H_SCoeff_ Prelaunch	Static	float64 array (16 values)	Band 5 prelaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B5H_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 5 postlaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B5H_SCoeff_Current	Dynamic	float64 array (16 values)	Band 5 current high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B6H_SCoeff_ Prelaunch	Static	float64 array (8 values)	Band 6 prelaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B6H_SCoeff_ Postlaunch	Static	float64 array (8 values)	Band 6 postlaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B6H_SCoeff_Current	Dynamic	float64 array (8 values)	Band 6 current high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B6H_SCoeffOff_ Prelaunch	Static	float64 array (8 values)	Band 6 prelaunch offset calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B6H_SCoeffOff_ Postlaunch	Static	float64 array (8 values)	Band 6 postlaunch offset calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B6H_SCoeffOff_ Current	Dynamic	float64 array (8 values)	Band 6 current offset calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B7H_SCoeff_ Prelaunch	Static	float64 array (16 values)	Band 7 prelaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B7H_SCoeff_ Postlaunch	Static	float64 array (16 values)	Band 7 postlaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B7H_SCoeff_Current	Dynamic	float64 array (16 values)	Band 7 current high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B8H_SCoeff_ Prelaunch	Static	float64 array (32 values)	Band 8 prelaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B8H_SCoeff_ Postlaunch	Static	float64 array (32 values)	Band 8 postlaunch high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Postlaunch
GROUP: SENSITIVITY_TEMPERATURES GROUP: SENSITIVITY_HIGH	B8H_SCoeff_Current	Dynamic	float64 array (32 values)	Band 8 current high-gain calibration temperature sensitivity coefficient Valid format: SNNN.NNNN, where S = "+" or "-" and NNN.NNNN = TBS	No	Yes	Thermal vacuum
GROUP: LAMP_RADIANCE GROUP: TRENDING_COEFFS	Lamp1_Coeffs	Static	float32 array (2 values)	Time since launch coefficients for Lamp 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: LAMP_RADIANCE GROUP: TRENDING_COEFFS	Lamp2_Coeffs	Static	float32 array (2 values)	Time since launch coefficients for Lamp 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B1L_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 1 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 off; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B1L_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 1 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 off; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B1L_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 1 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 off; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B1L_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 1 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B1L_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 1 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B1L_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 1 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B1L_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 1 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B1L_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 1 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B1L_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 1 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B2L_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 2 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 off; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B2L_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 2 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 off; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B2L_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 2 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 off; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B2L_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 2 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B2L_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 2 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B2L_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 2 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B2L_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 2 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B2L_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 2 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B2L_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 2 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B3L_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 3 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B3L_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 3 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B3L_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 3 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B3L_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 3 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B3L_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 3 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B3L_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 3 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B3L_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 3 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B3L_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 3 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B3L_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 3 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B4L_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 4 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B4L_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 4 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B4L_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 4 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B4L_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 4 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B4L_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 4 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B4L_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 4 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B4L_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 4 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B4L_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 4 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B4L_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 4 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B5L_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 5 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B5L_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 5 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B5L_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 5 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B5L_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 5 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B5L_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 5 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B5L_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 5 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B5L_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 5 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B5L_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 5 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B5L_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 5 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B7L_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 7 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B7L_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 7 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B7L_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 7 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B7L_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 7 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B7L_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 7 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B7L_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 7 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B7L_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 7 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B7L_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 7 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B7L_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 7 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B8L_Rad_State1_ Prelaunch	Static	float32 array (32 values)	Band 8 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B8L_Rad_State1_ Postlaunch	Static	float32 array (32 values)	Band 8 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B8L_Rad_State1_ Current	Dynamic	float32 array (32 values)	Band 8 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B8L_Rad_State2_ Prelaunch	Static	float32 array (32 values)	Band 8 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B8L_Rad_State2_ Postlaunch	Static	float32 array (32 values)	Band 8 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B8L_Rad_State2_ Current	Dynamic	float32 array (32 values)	Band 8 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B8L_Rad_State3_ Prelaunch	Static	float32 array (32 values)	Band 8 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B8L_Rad_State3_ Postlaunch	Static	float32 array (32 values)	Band 8 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_LOW	B8L_Rad_State3_ Current	Dynamic	float32 array (32 values)	Band 8 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; low-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B1H_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 1 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 off; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B1H_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 1 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 off; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B1H_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 1 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 off; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B1H_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 1 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B1H_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 1 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B1H_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 1 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B1H_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 1 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B1H_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 1 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B1H_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 1 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B2H_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 2 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B2H_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 2 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B2H_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 2 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B2H_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 2 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B2H_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 2 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B2H_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 2 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B2H_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 2 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B2H_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 2 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B2H_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 2 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B3H_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 3 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B3H_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 3 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B3H_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 3 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B3H_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 3 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B3H_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 3 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B3H_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 3 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B3H_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 3 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B3H_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 3 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B3H_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 3 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B4H_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 4 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B4H_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 4 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B4H_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 4 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B4H_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 4 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B4H_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 4 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B4H_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 4 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B4H_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 4 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B4H_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 4 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B4H_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 4 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B5H_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 5 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B5H_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 5 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B5H_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 5 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B5H_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 5 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B5H_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 5 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B5H_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 5 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B5H_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 5 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B5H_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 5 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B5H_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 5 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B7H_Rad_State1_ Prelaunch	Static	float32 array (16 values)	Band 7 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B7H_Rad_State1_ Postlaunch	Static	float32 array (16 values)	Band 7 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B7H_Rad_State1_ Current	Dynamic	float32 array (16 values)	Band 7 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B7H_Rad_State2_ Prelaunch	Static	float32 array (16 values)	Band 7 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B7H_Rad_State2_ Postlaunch	Static	float32 array (16 values)	Band 7 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B7H_Rad_State2_ Current	Dynamic	float32 array (16 values)	Band 7 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B7H_Rad_State3_ Prelaunch	Static	float32 array (16 values)	Band 7 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B7H_Rad_State3_ Postlaunch	Static	float32 array (16 values)	Band 7 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B7H_Rad_State3_ Current	Dynamic	float32 array (16 values)	Band 7 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B8H_Rad_State1_ Prelaunch	Static	float32 array (32 values)	Band 8 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B8H_Rad_State1_ Postlaunch	Static	float32 array (32 values)	Band 8 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B8H_Rad_State1_ Current	Dynamic	float32 array (32 values)	Band 8 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 1 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B8H_Rad_State2_ Prelaunch	Static	float32 array (32 values)	Band 8 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B8H_Rad_State2_ Postlaunch	Static	float32 array (32 values)	Band 8 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B8H_Rad_State2_ Current	Dynamic	float32 array (32 values)	Band 8 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 2 - lamp 1 off, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B8H_Rad_State3_ Prelaunch	Static	float32 array (32 values)	Band 8 prelaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B8H_Rad_State3_ Postlaunch	Static	float32 array (32 values)	Band 8 postlaunch internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	Postlaunch
GROUP: LAMP_RADIANCE GROUP: LAMP_RADIANCE_HIGH	B8H_Rad_State3_ Current	Dynamic	float32 array (32 values)	Band 8 current internal calibrator lamp effective spectral radiance in W/m^2-ster-µm; State 3 - lamp 1 on, lamp 2 on; high-gain mode Valid format: NNN.NNN, where NNN.NNN = TBS	No	Yes	AC02/ AC48

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: LAMP_REFERENCE	Lmp_Rtemp_ PreLaunch	Static	float32 array (14 values)	Prelaunch internal calibrator lamp radiance reference temperatures in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS T1 = Cal shutter flag temp T2 = Backup shutter flag temp T3 = Silicon focal plane array temp T4 = Cold focal plane monitor temp T5 = Cal lamp housing temp T6 = Scan line corrector temp T7 = Cal shutter hub temp T8 = Ambient pre-amp temp (high) T9 = Ambient pre-amp temp (low) T10 = Cold pre-amp temp (B7) T11 = Post-amp temp (B4) T12 = Primary mirror amp temp T13 = Secondary mirror temp T14 = Pan band post-amp temp	No	No	LPSO
GROUP: LAMP_REFERENCE	Lmp_Rtemp_ Postlaunch	Static	float32 array (14 values)	Postlaunch internal calibrator lamp radiance reference temperatures in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS Descriptions of T1 through T14 same as above	No	Yes	LPSO
GROUP: LAMP_REFERENCE	Lmp_Rtemp_Current	Dynamic	float32 array (14 values)	Current internal calibrator lamp radiance reference temperatures in degrees C Valid format: SNNN.NNN, where S = "+" or "-" and NNN.NNN = TBS Descriptions of T1 through T14 same as above	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B1L_Coefficients_ Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 1, low gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_ Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_ Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_ Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_ Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_ Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_ Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B2L_Coefficients_Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 2, low gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B3L_Coefficients_ Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 3, low gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B4L_Coefficients_ Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 4, low gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B5L_Coefficients_ Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 5, low gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B7L_Coefficients_ Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 7, low gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients _Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector17	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 17 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector18	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 18 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector19	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 19 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector20	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 20 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector21	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 21 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector22	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 22 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector23	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 23 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector24	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 24 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector25	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 25 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_ Detector26	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 26 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_Detector27	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 27 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_Detector28	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 28 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_Detector29	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 29 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_Detector30	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 30 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_Detector31	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 31 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_LOW	B8L_Coefficients_Detector32	Dynamic	float32 array (18 values)	IC coefficients for band 8, low gain, detector 32 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B1H_Coefficients_Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 1, high gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B2H_Coefficients_Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 2, high gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B3H_Coefficients_Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 3, high gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B4H_Coefficients_Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 4, high gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B5H_Coefficients_Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 5, high gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 6 Valid format: SNNN.NNNNNNN, where = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B7H_Coefficients_Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 7, high gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector1	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector2	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector3	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector4	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector5	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector6	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector7	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector8	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector9	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 9 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector10	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 10 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector11	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 11 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector12	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 12 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector13	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 13 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector14	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 14 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector15	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 15 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector16	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 16 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector17	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 17 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector18	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 18 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector19	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 19 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector20	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 20 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector21	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 21 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector22	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 22 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector23	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 23 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector24	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 24 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector25	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 25 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector26	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 26 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector27	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 27 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector28	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 28 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector29	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 29 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector30	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 30 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector31	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 31 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: REFLECTIVE_IC_COEFFS GROUP: REFLECT_IC_COEFFS_HIGH	B8H_Coefficients_Detector32	Dynamic	float32 array (18 values)	IC coefficients for band 8, high gain, detector 32 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	Yes	LPSO
GROUP: B6_VIEW_COEFFS	B6_View_Coefficients_Detector1	Static	float32 array (15 values)	View factor coefficients for band 6, detector 1 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS a1 = Scan line corrector view factor a2 = Central baffles (heater) a3 = Secondary mirror and mask view factor a4 = Primary mirror and mask view factor a5 = Scan mirror view factor a6 = Black body (isolated) view factor a7 = Black body (control) view factor a8 = Cold focal plane control view factor a9 = Cold focal plane monitor view factor a10 = Baffle (tube) view factor a11 = Baffle (support) view factor a12 = Telescope housing view factor frb = Integrated instrument view factor Vbb = Blocked aperture black body view factor Vsh = Blocked aperture shutter view factor	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: B6_VIEW_COEFFS	B6_View_Coefficients_ Detector2	Static	float32 array (15 values)	View factor coefficients for band 6, detector 2 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS Descriptions of the 15 coefficients are same as above	No	Yes	LPSO
GROUP: B6_VIEW_COEFFS	B6_View_Coefficients_ Detector3	Static	float32 array (15 values)	View factor coefficients for band 6, detector 3 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS Descriptions of the 15 coefficients are same as above	No	Yes	LPSO
GROUP: B6_VIEW_COEFFS	B6_View_Coefficients_ Detector4	Static	float32 array (15 values)	View factor coefficients for band 6, detector 4 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS Descriptions of the 15 coefficients are same as above	No	Yes	LPSO
GROUP: B6_VIEW_COEFFS	B6_View_Coefficients_ Detector5	Static	float32 array (15 values)	View factor coefficients for band 6, detector 5 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS Descriptions of the 15 coefficients are same as above	No	Yes	LPSO
GROUP: B6_VIEW_COEFFS	B6_View_Coefficients_ Detector6	Static	float32 array (15 values)	View factor coefficients for band 6, detector 6 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS Descriptions of the 15 coefficients are same as above	No	Yes	LPSO
GROUP: B6_VIEW_COEFFS	B6_View_Coefficients_ Detector7	Static	float32 array (15 values)	View factor coefficients for band 6, detector 7 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS Descriptions of the 15 coefficients are same as above	No	Yes	LPSO

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: B6_VIEW_COEFFS	B6_View_Coefficients_ Detector8	Static	float32 array (15 values)	View factor coefficients for band 6, detector 8 Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS Descriptions of the 15 coefficients are same as above	No	Yes	LPSO
GROUP: B6_TEMP_MODEL_COEFFS	B6_Temp_Model_Parm	Dynamic	float32 array (6 values)	Coefficients used to calculate scan mirror temperature where (a1) = Scan mirror/secondary mirror adjustment factor, (a2) = Average secondary mirror temperature, and (a3) - (a6) = reserved Valid format: SNNN.NNNNNNN, where S = "+" or "-" and SNNN.NNNNNNN = +1.0 (a1) SNNN.NNNNNNN = +0.0 (a2) SNNN.NNNNNNN = +0.0 (a3) SNNN.NNNNNNN = +0.0 (a4) SNNN.NNNNNNN = +0.0 (a5) SNNN.NNNNNNN = +0.0 (a6)	No	Yes	LPSO
GROUP: THERMISTOR_COEFFS	Black_Body_Isolated_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Black_Body_Control_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Cold_FP_Control_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Cold_FP_Monitor_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: THERMISTOR_COEFFS	Cal_Shutter_Flag_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Backup_Shutter_Flag_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Baffle_Heater_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Silicon_FP_Array_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Primary_Mirror_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Secondary_Mirror_Tem p	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Scan_Line_Corrector_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Baffle3_Tube_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Baffle2_Support_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Cal_Lamp_Housing_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: THERMISTOR_COEFFS	Cal_Shutter_Hub_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Ambient_Preamp_ HighCh_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Ambient_Preamp_ LowCh_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Postamp_Temp_B4	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Cold_Preamp_B7_Tem p	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Pan_Band_Postamp_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Telescope_Housing_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Primary_Mirror_Mask_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Secondary_Mirror_ Mask_Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Telescope_Baseplate_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS

Parameter Groups	Parameter Name	Value Type	Data Type	Description	LPS Need	LPG Need	Prelaunch Source
GROUP: THERMISTOR_COEFFS	Mux1_Power_Supply_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data (1) Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: THERMISTOR_COEFFS	Mux1_Electronics_ Temp	Static	float32 array (6 values)	Calibration coefficients for raw data (2) Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: LAMP_CURRENTS	Tec_Lamp_i1	Static	float32 array (2 values)	Calibration coefficients for raw data (3) Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: LAMP_CURRENTS	Tec_Lamp_i2	Static	float32 array (2 values)	Calibration coefficients for raw data (4) Valid format: SNNN.NNNNNNN, where S = + or - and NNN.NNNNNNN = TBS	No	No	SBRS
GROUP: FILL_PATTERNS	Band_Fill_Pattern	Static	uint8 array (2 values)	Fill pattern used by LPS for filling erroneous or missing image data minor frames Valid format: NNN, where NNN = (0, 255) (alternating 0, 255's)	Yes	Yes	LPSO

Notes: 1. Telemetry value contains the power supply temperature for "active" Mux, which could be either Mux 1 or Mux 2.

2. Telemetry value contains the electronics temperature for "active" Mux, which could be either Mux 1 or Mux 2.

3. Telemetry value contains current (mA) of primary on-board calibration lamp (telemetry name = TECLAMP1I)

4. Telemetry value contains current (mA) of secondary on-board calibration lamp (telemetry name = TECLAMP2I)

Section 3. CPF ODL

3.1 Introduction

The ODL syntax employs the following conventions:

- Parameter definition is in the form of parameter = value.
- Value can be either a scaler or an array. Array values are enclosed in parentheses and are separated by commas.
- Parameter arrays can and do exist on multiple lines.
- A carriage return <CR> and line feed <LF> end each line in the file.
- Blank spaces and lines are ignored.
- Each line of comments must begin with /* and end with */, including comments embedded on the same line as a parameter definition.
- Quotation marks are required for values that are text strings, including single characters. The exceptions to this rule are the GROUP and END_GROUP identifiers or values, which do not use quotation marks. The first two parameters in the file, Effective_Date_Begin and Effective_Date_End, also do not have quotations. ODL recognizes dates if they follow prescribed formats.
- Case is not significant, but uppercase is used for GROUP names to aid in readability.
- Indentation is not significant but is used for readability.
- The reserve word END concludes the file.

Unavailable parameter values are denoted by TBS (to be supplied). Most will be derived during prelaunch instrument and spacecraft testing and analysis. Formats for TBS numerical parameters are accurate; however, negative signs are not explicitly stated. A data dictionary that declares each parameter's data type and value range must be defined.

3.2 ODL Construct

```
GROUP = FILE_ATTRIBUTES
     Effective_Date_Begin      = YYYY-MM-DD
     Effective_Date_End        = YYYY-MM-DD
     CPF_File_Name     = -L7CPFyyyymmdd_yyyymmdd.nnî
END_GROUP = FILE_ATTRIBUTES

GROUP = EARTH_CONSTANTS
     Ellipsoid_Name    = -WGS84î
     Semi_Major_Axis   = 6378137.000
     Semi_Minor_Axis   = 6356752.314
     Ellipticity       =  0.0033528107
     Eccentricity      =  0.00669437999013
     Earth_Spin_Rate   = 72.92115855E-06
     Gravity_Constant  = 3.986005E14
```

```
     J2_Earth_Model_Term      = 1082.64E-06
END_GROUP = EARTH_CONSTANTS

GROUP = ORBIT_PARAMETERS
     WRS_Cycle_Days           = 16
     WRS_Cycle_Orbits         = 233
     Scenes_Per_Orbit         = 248
     Orbital_Period           = 5933.0472
     Angular_Momentum         = 53.104278E9
     Orbit_Radius             = 7083.437
     Orbit_Semimajor_Axis             = 7077.900
     Orbit_Semiminor_Axis             = 7069.580
     Orbit_Eccentricity               = 0.00118
     Inclination_Angle        = 98.2098
     Argument_Of_Perigee              = 90.0
     Descending_Node_Row              = 60
     Long_Path1_Row60         = -64.6
     Descending_Node_Time_Min  = "09:45"
     Descending_Node_Time_Max  = "10:00"
     Nodal_Regression_Rate            = 0.9856473
END_GROUP = ORBIT_PARAMETERS

GROUP = SCANNER_PARAMETERS
     Lines_Per_Scan_30 = 16
     Lines_Per_Scan_60 = 8
     Lines_Per_Scan_15 = 32
     Scans_Per_Scene   = 375
     Swath_Angle       = 0.26868
     Scan_Rate  = 2.21095
     Dwell_Time_30     = 9.6109603
     Dwell_Time_60     = 19.2220000
     Dwell_Time_15     = 4.8060000
     IC_Line_Length_30 = 1100
     IC_Line_Length_60 = 550
     IC_Line_Length_15 = 2200
     Scan_Line_Length_30      = 6330
     Scan_Line_Length_60      = 3165
     Scan_Line_Length_15      = 12660
     Filter_Frequency_30      = 52.02
     Filter_Frequency_60      = 26.01
     Filter_Frequency_15      = 115.0
     IFOV_B1234 = 42.5000
     IFOV_B57_along_scan      = 42.5000
     IFOV_B57_across_scan     = 42.5000
     IFOV_B6  = 85.0000
     IFOV_B8_along_scan       = 18.5000
     IFOV_B8_across_scan      = 21.25
     Scan_Period       = 142.92200
     Scan_Frequency    = 6.9968
     Active_Scan_Time  = 60743.013
     Turn_Around_Time  = 10.719
END_GROUP = SCANNER_PARAMETERS

GROUP = SPACECRAFT_PARAMETERS
     ADS_Interval      = 2.0
     ADS_Roll_Offset   = 0.375
     ADS_Yaw_Offset    = 0.875
     ADS_Pitch_Offset  = 1.375
     Data_Rate  = 74.903
END_GROUP = SPACECRAFT_PARAMETERS

GROUP = MIRROR_PARAMETERS
     GROUP = ANGLES_SME1_SAM
       Forward_Along_SME1_SAM  = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by SBRS */
       Forward_Cross_SME1_SAM  = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by SBRS */
       Forward_Angle1_SME1_SAM = 67171.0
       Forward_Angle2_SME1_SAM = 67159.0
       Reverse_Along_SME1_SAM  = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by SBRS */
       Reverse_Cross_SME1_SAM  = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by SBRS */
       Reverse_Angle1_SME1_SAM = 67159.0
       Reverse_Angle2_SME1_SAM = 67171.0
     END_GROUP = ANGLES_SME1_SAM
     GROUP = ANGLES_SME2_SAM
```

```
      Forward_Along_SME2_SAM         = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Forward_Cross_SME2_SAM         = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Forward_Angle1_SME2_SAM = 67182.0
      Forward_Angle2_SME2_SAM = 67160.0
      Reverse_Along_SME2_SAM         = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Reverse_Cross_SME2_SAM         = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Reverse_Angle1_SME2_SAM = 67160.0
      Reverse_Angle2_SME2_SAM = 67182.0
    END_GROUP = ANGLES_SME2_SAM
    GROUP = ANGLES_SME1_BUMP
      Forward_Along_SME1_Bump        = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Forward_Cross_SME1_Bump        = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Forward_Angle1_SME1_Bump       = 67171.0
      Forward_Angle2_SME1_Bump       = 67159.0
      Reverse_Along_SME1_Bump        = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Reverse_Cross_SME1_Bump        = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Reverse_Angle1_SME1_Bump       = 67159.0
      Reverse_Angle2_SME1_Bump       = 67171.0
    END_GROUP = ANGLES_SME1_BUMP
    GROUP = ANGLES_SME2_BUMP
      Forward_Along_SME2_Bump        = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Forward_Cross_SME2_Bump        = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Forward_Angle1_SME2_Bump       = 67182.0
      Forward_Angle2_SME2_Bump       = 67162.0
      Reverse_Along_SME2_Bump        = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Reverse_Cross_SME2_Bump        = (SN.NNNNESN, SN.NNNNESN,......)    /* 6 values TBS by
SBRS */
      Reverse_Angle1_SME2_Bump       = 67160.0
      Reverse_Angle2_SME2_Bump       = 67182.0
    END_GROUP = ANGLES_SME2_BUMP
    Error_Conversion_Factor          = 0.18845
END_GROUP = MIRROR_PARAMETERS

GROUP = SCAN_LINE_CORRECTOR
    Primary_Angular_Velocity         = 0.00966
    Secondary_Angular_Velocity       = 0.00960
    Primary_Corrector_Motion         = (N.NNNNN, N.NNNNN,......)   /* 6 values TBS by SBRS */
    Secondary_Corrector_Motion       = (N.NNNNN, N.NNNNN,......)   /* 6 values TBS by SBRS */
END_GROUP = SCAN_LINE_CORRECTOR

GROUP = FOCAL_PLANE_PARAMETERS
    GROUP = BAND_OFFSETS
      Along_Scan_Band_Offsets        = (NNNN.NNN, NNNN.NNN,......) /* 8 values TBS by SBRS */
      Across_Scan_Band_Offsets             = (NNNN.NNN, NNNN.NNN,......) /* 8 values TBS by
SBRS */
      Forward_Focal_Plane_Offsets    = (NNNN.NNN, NNNN.NNN,......) /* 8 values TBS by SBRS */
      Reverse_Focal_Plane_Offsets    = (NNNN.NNN, NNNN.NNN,......) /* 8 values TBS by SBRS */
    END_GROUP = BAND_OFFSETS
    GROUP = DETECTOR_OFFSETS
      Forward_Along_Scan_DO_B1       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Along_Scan_DO_B1       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Along_Scan_DO_B2       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Along_Scan_DO_B2       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Along_Scan_DO_B3       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Along_Scan_DO_B3       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Along_Scan_DO_B4       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Along_Scan_DO_B4       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Along_Scan_DO_B5       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Along_Scan_DO_B5       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Along_Scan_DO_B6       = (NNN.NNN, NNN.NNN,......)   /* 8 values TBS by SBRS */
      Reverse_Along_Scan_DO_B6       = (NNN.NNN, NNN.NNN,......)   /* 8 values TBS by SBRS */
      Forward_Along_Scan_DO_B7       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
```

```
      Reverse_Along_Scan_DO_B7       = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Along_Scan_DO_B8       = (NNN.NNN, NNN.NNN,......)   /* 32 values TBS by SBRS */
      Reverse_Along_Scan_DO_B8       = (NNN.NNN, NNN.NNN,......)   /* 32 values TBS by SBRS */
      Forward_Across_Scan_DO_B1      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Across_Scan_DO_B1      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Across_Scan_DO_B2      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Across_Scan_DO_B2      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Across_Scan_DO_B3      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Across_Scan_DO_B3      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Across_Scan_DO_B4      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Across_Scan_DO_B4      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Across_Scan_DO_B5      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Across_Scan_DO_B5      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Across_Scan_DO_B6      = (NNN.NNN, NNN.NNN,......)   /* 8 values TBS by SBRS */
      Reverse_Across_Scan_DO_B6      = (NNN.NNN, NNN.NNN,......)   /* 8 values TBS by SBRS */
      Forward_Across_Scan_DO_B7      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Reverse_Across_Scan_DO_B7      = (NNN.NNN, NNN.NNN,......)   /* 16 values TBS by SBRS */
      Forward_Across_Scan_DO_B8      = (NNN.NNN, NNN.NNN,......)   /* 32 values TBS by SBRS */
      Reverse_Across_Scan_DO_B8      = (NNN.NNN, NNN.NNN,......)   /* 32 values TBS by SBRS */
    END_GROUP = DETECTOR_OFFSETS
    GROUP = ODD_EVEN_OFFSETS
      Forward_Even_Detector_Shift    = (NNN.NNN, NNN.NNN,......)   /* 8 values TBS by SBRS */
      Forward_Odd_Detector_Shift     = (NNN.NNN, NNN.NNN,......)   /* 8 values TBS by SBRS */
      Reverse_Even_Detector_Shift    = (NNN.NNN, NNN.NNN,......)   /* 8 values TBS by SBRS */
      Reverse_Odd_Detector_Shift     = (NNN.NNN, NNN.NNN,......)   /* 8 values TBS by SBRS */
    END_GROUP = ODD_EVEN_OFFSETS
END_GROUP = FOCAL_PLANE_PARAMETERS

GROUP = ATTITUDE_PARAMETERS
    Gyro_To_Attitude_Matrix  = (1.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 1.0)     /* 9 values TBS
by SBRS */
    ADSA_To_ETM_Matrix               = (1.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 1.0)     /* 9
values TBS by SBRS */
    Attitude_To_ETM_Matrix   = (1.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 1.0)     /* 9 values TBS
by SBRS */
    Spacecraft_Roll_Bias             = N.NNNNNNN                  /* TBS by LMC */
    Spacecraft_Pitch_Bias            = N.NNNNNNN                  /* TBS by LMC */
    Spacecraft_Yaw_Bias              = N.NNNNNNN                  /* TBS by LMC */
END_GROUP = ATTITUDE_PARAMETERS

GROUP = TIME_PARAMETERS
    Scan_Time         = 60743.0
    Forward_First_Half_Time          = 30371.4
    Forward_Second_Half_Time  = 30371.6
    Reverse_First_Half_Time          = 30371.6
    Reverse_Second_Half_Time  = 30371.4
END_GROUP = TIME_PARAMETERS

GROUP = TRANSFER_FUNCTION
    GROUP = IMU
      Fn = 2.201             /* 1 value TBS by SBRS */
      Zeta   = 0.7022        /* 1 value TBS by SBRS */
      Tau    = 1.4468E-3             /* 1 value TBS by SBRS */
      P  = -3.259E-3         /* 1 value TBS by SBRS */
      Ak = 1.00121           /* 1 value TBS by SBRS */
    END_GROUP = IMU
    GROUP = ADS
      ADS_num  = (N.NNNNEN, N.NNNNEN,......) /* 18 values TBS by SBRS */
      ADS_den  = (N.NNNNEN, N.NNNNEN,......) /* 18 values TBS by SBRS */
      ADS_num_temp    = (N.NNNNEN, N.NNNNEN,......) /* 18 values TBS by SBRS */
      ADS_den_temp    = (N.NNNNEN, N.NNNNEN,......) /* 18 values TBS by SBRS */
    END_GROUP = ADS
    GROUP = PREFILTER
      ADSPre_W = (0.0, 0.0, 0.0, 0.0, 0.0)   /* 5 values TBS by SBRS */
      ADSPre_H = (0.0, 0.0, 0.0, 0.0, 0.0)   /* 5 values TBS by SBRS */
      ADSPre_T = (0.0, 0.0, 0.0, 0.0, 0.0)   /* 5 values TBS by SBRS */
    END_GROUP = PREFILTER
END_GROUP = TRANSFER_FUNCTION

GROUP = UT1_TIME_PARAMETERS
    UT1_Year   =(YYYY, YYYY,......)          /* 180 values TBS NEOS */
    UT1_Month         = ("MMM", "MMM",......)       /* 180 values TBS NEOS */
    UT1_Day    = (NN,NN......)               /* 180 values TBS NEOS */
```

```
     UT1_Modified_Julian              = (NNNNN,NNNNN,......) /* 180 values TBS NEOS */
     UT1_X      = (N.NNNNN, N.NNNNN,......)   /* 180 values TBS NEOS */
     UT1_Y      = (N.NNNNN, N.NNNNN,......)   /* 180 values TBS NEOS */
     UT1_UTC    = (N.NNNNN, N.NNNNN,......)   /* 180 values TBS NEOS */
END_GROUP = UT1_TIME_PARAMETERS

GROUP = DETECTOR_STATUS
     Status_Band1      = (NNNNN, NNNNN,......)       /* 16 values TBS by SBRS */
     Status_Band2      = (NNNNN, NNNNN,......)       /* 16 values TBS by SBRS */
     Status_Band3      = (NNNNN, NNNNN,......)       /* 16 values TBS by SBRS */
     Status_Band4      = (NNNNN, NNNNN,......)       /* 16 values TBS by SBRS */
     Status_Band5      = (NNNNN, NNNNN,......)       /* 16 values TBS by SBRS */
     Status_Band6      = (NNNNN, NNNNN,......)       /* 8 values TBS by SBRS */
     Status_Band7      = (NNNNN, NNNNN,......)       /* 16 values TBS by SBRS */
     Status_Band8      = (NNNNN, NNNNN,......)       /* 32 values TBS by SBRS */
END_GROUP = DETECTOR_STATUS

GROUP = DETECTOR_GAINS
     GROUP = DETECTOR_GAINS_LOW
       B1L_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B1L_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B1L_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B2L_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B2L_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B2L_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B3L_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B3L_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B3L_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B4L_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B4L_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B4L_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B5L_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B5L_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B5L_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B6L_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B6L_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B6L_Current     = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B7L_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B7L_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B7L_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B8L_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 32 values TBS by SBRS */
       B8L_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 32 values TBS by SBRS */
       B8L_Current     = (NNN.NNNN, NNN.NNNN,......) /* 32 values TBS by SBRS */
     END_GROUP = DETECTOR_GAINS_LOW
     GROUP = DETECTOR_GAINS_HIGH
       B1H_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B1H_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B1H_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B2H_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B2H_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B2H_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B3H_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B3H_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B3H_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B4H_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B4H_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B4H_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B5H_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B5H_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B5H_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B6H_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B6H_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B6H_Current     = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B7H_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B7H_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B7H_Current     = (NNN.NNNN, NNN.NNNN,......) /* 16 values TBS by SBRS */
       B8H_Prelaunch   = (NNN.NNNN, NNN.NNNN,......) /* 32 values TBS by SBRS */
       B8H_Postlaunch  = (NNN.NNNN, NNN.NNNN,......) /* 32 values TBS by SBRS */
       B8H_Current     = (NNN.NNNN, NNN.NNNN,......) /* 32 values TBS by SBRS */
     END_GROUP = DETECTOR_GAINS_HIGH
GROUP = DETECTOR_GAINS

GROUP = BIAS_LOCATIONS
```

```
     Forward_Bias_Location_30  = NNN   /* TBS by LPSO */
     Forward_Bias_Length_30    = NNN   /* TBS by LPSO */
     Forward_IC_Region_30              = NNN   /* TBS by LPSO */
     Reverse_Bias_Location_30  = NNN   /* TBS by LPSO */
     Reverse_Bias_Length_30    = NNN   /* TBS by LPSO */
     Reverse_IC_Region_30              = NNN   /* TBS by LPSO */
     Forward_Bias_Location_60  = NNN   /* TBS by LPSO */
     Forward_Bias_Length_60    = NNN   /* TBS by LPSO */
     Forward_IC_Region_60              = NNN   /* TBS by LPSO */
     Reverse_Bias_Location_60  = NNN   /* TBS by LPSO */
     Reverse_Bias_Length_60    = NNN   /* TBS by LPSO */
     Reverse_IC_Region_60              = NNN   /* TBS by LPSO */
     Forward_Bias_Location_15  = NNN   /* TBS by LPSO */
     Forward_Bias_Length_15    = NNN   /* TBS by LPSO */
     Forward_IC_Region_15              = NNN   /* TBS by LPSO */
     Reverse_Bias_Location_15  = NNN   /* TBS by LPSO */
     Reverse_Bias_Length_15    = NNN   /* TBS by LPSO */
     Reverse_IC_Region_15              = NNN   /* TBS by LPSO */
END_GROUP = BIAS_LOCATIONS

GROUP = DETECTOR_BIASES_B6
     GROUP = DETECTOR_BIASES_B6_LOW
       B6L_Bias_Prelaunch      = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B6L_Bias_Postlaunch     = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B6L_Bias_Current        = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
     END_GROUP = DETECTOR_BIASES_B6_LOW
     GROUP = DETECTOR_BIASES_B6_HIGH
       B6H_Bias_Prelaunch      = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B6H_Bias_Postlaunch     = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
       B6H_Bias_Current        = (NNN.NNNN, NNN.NNNN,......) /* 8 values TBS by SBRS */
     END_GROUP = DETECTOR_BIASES__B6_HIGH
END_GROUP = DETECTOR_BIASES_B6

GROUP = ACCA_BIASES
     GROUP = ACCA_BIASES_LOW
       B1L_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B2L_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B3L_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B4L_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B5L_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B6L_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 8 values TBS by LPSO */
       B7L_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B8L_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 32 values TBS by LPSO */
     END_GROUP = ACCA_BIASES_LOW
     GROUP = ACCA_BIASES_HIGH
       B1H_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B2H_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B3H_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B4H_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B5H_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B6H_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 8 values TBS by LPSO */
       B7H_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 16 values TBS by LPSO */
       B8H_ACCA_Bias   = (NNN.NNNN, NN.NNNN,......)  /* 32 values TBS by LPSO */¤
     END_GROUP = ACCA_BIASES_HIGH
END_GROUP = ACCA_BIASES

GROUP = ACCA_THRESHOLDS
     Thresh_B3  = 0.3000
     Thresh_B3_Lower   = 0.06
     Thresh_B56_High   = 225.000
     Thresh_B56_Low    = 210.000
     Thresh_B6  = 300.000
     Thresh_B45_Ratio  = 1.0750
     Thresh_B42_Ratio  = 2.0000
     Thresh_B43_Ratio  = 2.0000
     Thresh_NDSI_Max   = 0.7000
     Thresh_NDSI_Min   = -0.2500
     Thresh_NDSI_Snow  = 0.8000
     Cloud_Percent_Min = 0.4000
     Desert_Index      = 0.5000
     Thresh_Snow_Percent       = 1.000
     Thermal_Effect_High       = 40.000
     Thermal_Effect_Low        = 30.000
```

```
     B6Max_Maxthresh_Diff      = 2.000
END_GROUP = ACCA_THRESHOLDS

GROUP = SOLAR_SPECTRAL_IRRADIANCES
     B1_Solar_Irradiance       = 1957.000
     B2_Solar_Irradiance       = 1829.000
     B3_Solar_Irradiance       = 1557.000
     B4_Solar_Irradiance       = 1047.000
     B5_Solar_Irradiance       = 219.300
     B7_Solar_Irradiance       = 74.520
     B8_Solar_Irradiance       = (NNNN.NNN)   /* 1 value TBS by LPSO */
END_GROUP = SOLAR_SPECTRAL_IRRADIANCES

GROUP = THERMAL_CONSTANTS
     K1_Constant       = 607.760
     K2_Constant       =  1260.560
END_GROUP = THERMAL_CONSTANTS

GROUP = SCALING_PARAMETERS
     GROUP = SCALING_PARAMETERS_LOW
       B1L_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B2L_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B3L_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B4L_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B5L_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B6L_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B7L_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B8L_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
     END_GROUP = SCALING_PARAMETERS_LOW
     GROUP = SCALING_PARAMETERS_HIGH
       B1H_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B2H_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B3H_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B4H_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B5H_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B6H_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B7H_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
       B8H_Lmin_Lmax   = (SNN.NNNN, SNN.NNNN) /* 2 values TBS by LPSO */
     END_GROUP = SCALING_PARAMETERS_HIGH
END_GROUP = SCALING_PARAMETERS

GROUP = MTF_COMPENSATION
       B1_weights_along        = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B1_weights_across       = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B2_weights_along        = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B2_weights_across       = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B3_weights_along        = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B3_weights_across       = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B4_weights_along        = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B4_weights_across       = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B5_weights_along        = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B5_weights_across       = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B6_weights_along        = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B6_weights_across       = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B7_weights_along        = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B7_weights_across       = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B8_weights_along        = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
       B8_weights_across       = (SNN.NNNN, SNN.NNNN,......) /* 5 values TBS by LPSO */
END_GROUP = MTF_COMPENSATION

GROUP = MEMORY_EFFECT
     GROUP = ME_MAGNITUDES
       B1_ME_Magnitude = (NNN.NNNNNNN, NNN.NNNNNNN,...)      /* 16 values TBS by LPSO */
       B2_ME_Magnitude = (NNN.NNNNNNN, NNN.NNNNNNN,...)      /* 16 values TBS by LPSO */
       B3_ME_Magnitude = (NNN.NNNNNNN, NNN.NNNNNNN,...)      /* 16 values TBS by LPSO */
       B4_ME_Magnitude = (NNN.NNNNNNN, NNN.NNNNNNN,...)      /* 16 values TBS by LPSO */
       B5_ME_Magnitude = (NNN.NNNNNNN, NNN.NNNNNNN,...)      /* 16 values TBS by LPSO */
       B6_ME_Magnitude = (NNN.NNNNNNN, NNN.NNNNNNN,...)      /* 8 values TBS by LPSO */
       B7_ME_Magnitude = (NNN.NNNNNNN, NNN.NNNNNNN,...)      /* 16 values TBS by LPSO */
       B8_ME_Magnitude = (NNN.NNNNNNN, NNN.NNNNNNN,...)      /* 32 values TBS by LPSO */
     END_GROUP = ME_EFFECT_MAGNITUDES
     GROUP = ME_TIME_CONSTANTS
```

```
      B1_ME_Time_Constant            = (NNNN.NNNNNNN, NNNN.NNNNNNN,..)    /* 16 values TBS by
LPSO */
      B2_ME_Time_Constant            = (NNNN.NNNNNNN, NNNN.NNNNNNN,..)    /* 16 values TBS by
LPSO */
      B3_ME_Time_Constant            = (NNNN.NNNNNNN, NNNN.NNNNNNN,..)    /* 16 values TBS by
LPSO */
      B4_ME_Time_Constant            = (NNNN.NNNNNNN, NNNN.NNNNNNN,..)    /* 16 values TBS by
LPSO */
      B5_ME_Time_Constant            = (NNNN.NNNNNNN, NNNN.NNNNNNN,..)    /* 16 values TBS by
LPSO */
      B6_ME_Time_Constant            = (NNNN.NNNNNNN, NNNN.NNNNNNN,..)    /* 8 values TBS by
LPSO */
      B7_ME_Time_Constant            = (NNNN.NNNNNNN, NNNN.NNNNNNN,..)    /* 16 values TBS by
LPSO */
      B8_ME_Time_Constant            = (NNNN.NNNNNNN, NNNN.NNNNNNN,..)    /* 32 values TBS by
LPSO */
    END_GROUP = ME_TIME_CONSTANTS
END_GROUP = MEMORY_EFFECT

GROUP = GHOST_PULSE
    Ghost_Pulse_Endpoints    = (NNNN.NNNN, NNNN.NNNN)      /* 2 values TBS by LPSO */
END_GROUP = GHOST_PULSE

GROUP = SCAN_CORRELATED_SHIFT
    SCS_Reference_Detectors  = (NN, NN,......)     /*  7 values TBS by LPSO */
    GROUP = SCS_LOW
      B1L_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B2L_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B3L_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B4L_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B5L_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B7L_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B8L_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 32  values TBS by
LPSO */

    END_GROUP = SCS_LOW
    GROUP = SCS_HIGH
      B1H_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B2H_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B3H_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B4H_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B5H_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B7H_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 16 values TBS by
LPSO */
      B8H_SCS_Magnitudes             = (SNNN.NNNNNNN, SNN.NNNNNNN,....)   /* 32  values TBS by
LPSO */
    END_GROUP = SCS_HIGH
END_GROUP = SCAN_CORRELATED_SHIFT

GROUP = STRIPING
    GROUP = STRIPING_FLAG_LOW
      Correction_Reference_B1_Low    = N    /* 1 value TBS by LPSO */
      Correction_Reference_B2_Low    = N    /* 1 value TBS by LPSO */
      Correction_Reference_B3_Low    = N    /* 1 value TBS by LPSO */
      Correction_Reference_B4_Low    = N    /* 1 value TBS by LPSO */
      Correction_Reference_B5_Low    = N    /* 1 value TBS by LPSO */
      Correction_Reference_B6_Low    = N    /* 1 value TBS by LPSO */
      Correction_Reference_B7_Low    = N    /* 1 value TBS by LPSO */
      Correction_Reference_B8_Low    = N    /* 1 value TBS by LPSO */
    END_GROUP = STRIPING_FLAG_LOW
    GROUP = STRIPING_FLAG_HIGH
      Correction_Reference_B1_High   = N    /* 1 value TBS by LPSO */
```

```
      Correction_Reference_B2_High   = N     /* 1 value TBS by LPSO */
      Correction_Reference_B3_High   = N     /* 1 value TBS by LPSO */
      Correction_Reference_B4_High   = N     /* 1 value TBS by LPSO */
      Correction_Reference_B5_High   = N     /* 1 value TBS by LPSO */
      Correction_Reference_B6_High   = N     /* 1 value TBS by LPSO */
      Correction_Reference_B7_High   = N     /* 1 value TBS by LPSO */
      Correction_Reference_B8_High   = N     /* 1 value TBS by LPSO */
    END_GROUP = STRIPING_FLAG_HIGH
END_GROUP = STRIPING

GROUP = HISTOGRAM
    GROUP = DETECTOR_NOISE
    GROUP = DETECTOR_NOISE_LOW
      Detector_Noise_Level_B1_Low    = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B2_Low    = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B3_Low    = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B4_Low    = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B5_Low    = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B6_Low    = (NN.NNNN, NN.NNNN,......)   /* 8 values TBS by LPSO */
      Detector_Noise_Level_B7_Low    = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B8_Low    = (NN.NNNN, NN.NNNN,......)   /* 32 values TBS by LPSO */
    END_GROUP = DETECTOR_NOISE_LOW
    GROUP = DETECTOR_NOISE_HIGH
      Detector_Noise_Level_B1_High   = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B2_High   = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B3_High   = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B4_High   = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B5_High   = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B6_High   = (NN.NNNN, NN.NNNN,......)   /* 8 values TBS by LPSO */
      Detector_Noise_Level_B7_High   = (NN.NNNN, NN.NNNN,......)   /* 16 values TBS by LPSO */
      Detector_Noise_Level_B8_High   = (NN.NNNN, NN.NNNN,......)   /* 32 values TBS by LPSO */
    END_GROUP = DETECTOR_NOISE_HIGH
    END_GROUP = DETECTOR_NOISE
    GROUP = REFERENCE_DETECTORS
      Reference_Detector_B1   = NN   /* 1 value TBS by LPSO */
      Reference_Detector_B2   = NN   /* 1 value TBS by LPSO */
      Reference_Detector_B3   = NN   /* 1 value TBS by LPSO */
      Reference_Detector_B4   = NN   /* 1 value TBS by LPSO */
      Reference_Detector_B5   = NN   /* 1 value TBS by LPSO */
      Reference_Detector_B6   = NN   /* 1 value TBS by LPSO */
      Reference_Detector_B7   = NN   /* 1 value TBS by LPSO */
      Reference_Detector_B8   = NN   /* 1 value TBS by LPSO */
    END_GROUP = REFERENCE_DETECTORS
    GROUP = SATURATION_THRESHOLDS
      Saturation_Bin_Threshold_B1           = NNNNN /* 1 value TBS by LPSO */
      Saturation_Bin_Threshold_B2    = NNNNN /* 1 value TBS by LPSO */
      Saturation_Bin_Threshold_B3    = NNNNN /* 1 value TBS by LPSO */
      Saturation_Bin_Threshold_B4    = NNNNN /* 1 value TBS by LPSO */
      Saturation_Bin_Threshold_B5    = NNNNN /* 1 value TBS by LPSO */
      Saturation_Bin_Threshold_B6    = NNNNN /* 1 value TBS by LPSO */
      Saturation_Bin_Threshold_B7    = NNNNN /* 1 value TBS by LPSO */
      Saturation_Bin_Threshold_B8    = NNNNN /* 1 value TBS by LPSO */
    END_GROUP = SATURATION_THRESHOLDS
    GROUP = ADJACENT_BINS
    GROUP = BIN_NUMBER
      Adjacent_Bin_Number_B1  = 2
      Adjacent_Bin_Number_B2  = 2
      Adjacent_Bin_Number_B3  = 2
      Adjacent_Bin_Number_B4  = 2
      Adjacent_Bin_Number_B5  = 2
      Adjacent_Bin_Number_B6  = 2
      Adjacent_Bin_Number_B7  = 2
      Adjacent_Bin_Number_B8  = 2
    END_GROUP = BIN_NUMBER
    GROUP = BIN_THRESHOLD
      Adjacent_Bin_Threshold_B1      = 10
      Adjacent_Bin_Threshold_B2      = 10
      Adjacent_Bin_Threshold_B3      = 10
      Adjacent_Bin_Threshold_B4      = 10
      Adjacent_Bin_Threshold_B5      = 10
      Adjacent_Bin_Threshold_B6      = 10
      Adjacent_Bin_Threshold_B7      = 10
      Adjacent_Bin_Threshold_B8      = 10
```

```
    END_GROUP = BIN_THRESHOLD
    END_GROUP = ADJACENT_BINS
    GROUP = STARTING_PIXEL
      Start_pixel_B1  = NNN   /* 1 value TBS by LPSO */
      Start_pixel_B2  = NNN   /* 1 value TBS by LPSO */
      Start_pixel_B3  = NNN   /* 1 value TBS by LPSO */
      Start_pixel_B4  = NNN   /* 1 value TBS by LPSO */
      Start_pixel_B5  = NNN   /* 1 value TBS by LPSO */
      Start_pixel_B6  = NNN   /* 1 value TBS by LPSO */
      Start_pixel_B7  = NNN   /* 1 value TBS by LPSO */
      Start_pixel_B8  = NNN   /* 1 value TBS by LPSO */
    END_GROUP = STARTING_PIXEL
    GROUP = WINDOW_WIDTH
      Window_Samples_B1       = NNN   /* 1 value TBS by LPSO */
      Window_Samples_B2       = NNN   /* 1 value TBS by LPSO */
      Window_Samples_B3       = NNN   /* 1 value TBS by LPSO */
      Window_Samples_B4       = NNN   /* 1 value TBS by LPSO */
      Window_Samples_B5       = NNN   /* 1 value TBS by LPSO */
      Window_Samples_B6       = NNN   /* 1 value TBS by LPSO */
      Window_Samples_B7       = NNN   /* 1 value TBS by LPSO */
      Window_Samples_B8       = NNN   /* 1 value TBS by LPSO */
    END_GROUP = WINDOW_WIDTH
    GROUP = WINDOW_LENGTH
      Window_Scans_B1 = NNN   /* 1 value TBS by LPSO */
      Window_Scans_B2 = NNN   /* 1 value TBS by LPSO */
      Window_Scans_B3 = NNN   /* 1 value TBS by LPSO */
      Window_Scans_B4 = NNN   /* 1 value TBS by LPSO */
      Window_Scans_B5 = NNN   /* 1 value TBS by LPSO */
      Window_Scans_B6 = NNN   /* 1 value TBS by LPSO */
      Window_Scans_B7 = NNN   /* 1 value TBS by LPSO */
      Window_Scans_B8 = NNN   /* 1 value TBS by LPSO */
    END_GROUP = WINDOW_LENGTH
    GROUP = OVERLAPPING_SCANS
      Overlap_Scans_B1        = NNN   /* 1 value TBS by LPSO */
      Overlap_Scans_B2        = NNN   /* 1 value TBS by LPSO */
      Overlap_Scans_B3        = NNN   /* 1 value TBS by LPSO */
      Overlap_Scans_B4        = NNN   /* 1 value TBS by LPSO */
      Overlap_Scans_B5        = NNN   /* 1 value TBS by LPSO */
      Overlap_Scans_B6        = NNN   /* 1 value TBS by LPSO */
      Overlap_Scans_B7        = NNN   /* 1 value TBS by LPSO */
      Overlap_Scans_B8        = NNN   /* 1 value TBS by LPSO */
    GROUP = OVERLAPPING_SCANS
END_GROUP = HISTOGRAM

GROUP = IMPULSE_NOISE
      Median_Filter_Width    =  3
    GROUP = IN_THRESHOLD
      B1L_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B2L_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B3L_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B4L_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B5L_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B6L_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 8 values TBS by LPSO */
      B7L_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B8L_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 32 values TBS by LPSO */
      B1H_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B2H_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B3H_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B4H_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B5H_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B6H_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 8 values TBS by LPSO */
      B7H_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 16 values TBS by LPSO */
      B8H_Threshold   = (NN.NNNNNNN, NN.NNNNNNN,...)       /* 32 values TBS by LPSO */
    GROUP = IN_THRESHOLD
END_GROUP = IMPULSE_NOISE

GROUP = COHERENT_NOISE
    Frequency_Components     = 10
    GROUP = CN_FREQUENCY_PARAMETERS
      GROUP = FREQUENCY_MEANS
      B1_Frequency_Mean       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Frequency_Mean       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Frequency_Mean       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
```

```
      B4_Frequency_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Frequency_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Frequency_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Frequency_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Frequency_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = FREQUENCY_MEANS
    GROUP = FREQUENCY_SIGMAS
      B1_Frequency_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Frequency_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Frequency_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Frequency_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Frequency_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Frequency_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Frequency_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Frequency_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = FREQUENCY_MEANS
    GROUP = FREQUENCY_MINIMUMS
      B1_Frequency_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Frequency_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Frequency_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Frequency_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Frequency_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Frequency_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Frequency_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Frequency_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = FREQUENCY_MINIMUMS
    GROUP = FREQUENCY_MAXIMUMS
      B1_Frequency_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Frequency_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Frequency_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Frequency_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Frequency_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Frequency_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Frequency_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Frequency_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = FREQUENCY_MAXIMUMS
END_GROUP = CN_FREQUENCY_PARAMETERS
GROUP = CN_PHASE_PARAMETERS
    GROUP = PHASE_MEANS
      B1_Phase_Mean   = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Phase_Mean   = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Phase_Mean   = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Phase_Mean   = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Phase_Mean   = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Phase_Mean   = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Phase_Mean   = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Phase_Mean   = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = PHASE_MEANS
    GROUP = PHASE_SIGMAS
      B1_Phase_Sigma  = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Phase_Sigma  = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Phase_Sigma  = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Phase_Sigma  = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Phase_Sigma  = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Phase_Sigma  = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Phase_Sigma  = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Phase_Sigma  = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = PHASE_SIGMAS
    GROUP = PHASE_MINIMUMS
      B1_Phase_Min    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Phase_Min    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Phase_Min    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Phase_Min    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Phase_Min    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Phase_Min    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Phase_Min    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Phase_Min    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = PHASE_MINIMUMS
    GROUP = PHASE_MAXIMUMS
      B1_Phase_Max    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Phase_Max    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Phase_Max    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Phase_Max    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
```

```
      B5_Phase_Max    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Phase_Max    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Phase_Max    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Phase_Max    = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = PHASE_MAXIMUMS
END_GROUP = CN_PHASE_PARAMETERS
GROUP = CN_MAGNITUDE_PARAMETERS
    GROUP = MAGNITUDE_MEANS
      B1_Magnitude_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Magnitude_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Magnitude_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Magnitude_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Magnitude_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Magnitude_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Magnitude_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Magnitude_Mean      = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = MAGNITUDE_MEANS
    GROUP = MAGNITUDE_SIGMAS
      B1_Magnitude_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Magnitude_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Magnitude_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Magnitude_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Magnitude_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Magnitude_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Magnitude_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Magnitude_Sigma     = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = MAGNITUDE_SIGMAS
    GROUP = MAGNITUDE_MINIMUMS
      B1_Magnitude_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Magnitude_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Magnitude_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Magnitude_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Magnitude_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Magnitude_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Magnitude_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Magnitude_Min       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = MAGNITUDE_MINIMUMS
    GROUP = MAGNITUDE_MAXIMUMS
      B1_Magnitude_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B2_Magnitude_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B3_Magnitude_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B4_Magnitude_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B5_Magnitude_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B6_Magnitude_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B7_Magnitude_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
      B8_Magnitude_Max       = (N.NNNNNNN, N.NNNNNNN,......)      /* 10 values TBS by LPSO */
    END_GROUP = MAGNITUDE_MAXIMUMS
    GROUP = CN_MAGNITUDE_PARAMETERS
END_GROUP = COHERENT_NOISE

GROUP = DETECTOR_SATURATION
    GROUP = AD_CONVERTER_SATURATION
      GROUP = AD_CONVERTER_SATURATION_LOW
        High_AD_Level_B1_low  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B2_low  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B3_low  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B4_low  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B5_low  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B6_low  = (NNN, NNN,......)    /* 8 values TBS by SBRS */
        High_AD_Level_B7_low  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B8_low  = (NNN, NNN,......)    /* 32 values TBS by SBRS */
        Low_AD_Level_B1_low   = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B2_low   = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B3_low   = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B4_low   = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B5_low   = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B6_low   = (NNN, NNN,......)    /* 8 values TBS by SBRS */
        Low_AD_Level_B7_low   = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B8_low   = (NNN, NNN,......)    /* 32 values TBS by SBRS */
      END_GROUP = AD_CONVERTER_SATURATION_LOW
      GROUP = AD_CONVERTER_SATURATION_HIGH
        High_AD_Level_B1_high = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B2_high = (NNN, NNN,......)    /* 16 values TBS by SBRS */
```

```
        High_AD_Level_B3_high = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B4_high = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B5_high = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B6_high = (NNN, NNN,......)    /* 8 values TBS by SBRS */
        High_AD_Level_B7_high = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_AD_Level_B8_high = (NNN, NNN,......)    /* 32 values TBS by SBRS */
        Low_AD_Level_B1_high  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B2_high  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B3_high  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B4_high  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B5_high  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B6_high  = (NNN, NNN,......)    /* 8 values TBS by SBRS */
        Low_AD_Level_B7_high  = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_AD_Level_B8_high  = (NNN, NNN,......)    /* 32 values TBS by SBRS */
      END_GROUP = AD_CONVERTER_SATURATION_HIGH
    END_GROUP = AD_CONVERTER_SATURATION
    GROUP = ANALOG_SIGNAL_SATURATION
      GROUP = ANALOG_SIGNAL_SATURATION_LOW
        High_Analog_Level_B1_low     = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B2_low     = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B3_low     = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B4_low     = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B5_low     = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B6_low     = (NNN, NNN,......)    /* 8 values TBS by SBRS */
        High_Analog_Level_B7_low     = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B8_low     = (NNN, NNN,......)    /* 32 values TBS by SBRS */
        Low_Analog_Level_B1_low      = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B2_low      = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B3_low      = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B4_low      = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B5_low      = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B6_low      = (NNN, NNN,......)    /* 8 values TBS by SBRS */
        Low_Analog_Level_B7_low      = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B8_low      = (NNN, NNN,......)    /* 32 values TBS by SBRS */
      END_GROUP = ANALOG_SIGNAL_SATURATION_LOW
      GROUP = ANALOG_SIGNAL_SATURATION_HIGH
        High_Analog_Level_B1_high           = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B2_high    = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B3_high    = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B4_high    = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B5_high    = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B6_high    = (NNN, NNN,......)    /* 8 values TBS by SBRS */
        High_Analog_Level_B7_high    = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        High_Analog_Level_B8_high    = (NNN, NNN,......)    /* 32 values TBS by SBRS */
        Low_Analog_Level_B1_high            = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B2_high            = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B3_high            = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B4_high            = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B5_high            = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B6_high            = (NNN, NNN,......)    /* 8 values TBS by SBRS */
        Low_Analog_Level_B7_high            = (NNN, NNN,......)    /* 16 values TBS by SBRS */
        Low_Analog_Level_B8_high            = (NNN, NNN,......)    /* 32 values TBS by SBRS */
      END_GROUP = ANALOG_SIGNAL_SATURATION_HIGH
    END_GROUP = ANALOG_SIGNAL_SATURATION
END_GROUP = DETECTOR_SATURATION

GROUP = REFERENCE_TEMPERATURES
    GROUP = REFERENCE_LOW
      B1L_RTemp_Prelaunch           = SNNN.NNN     /* TBS by SBRS */
      B1L_RTemp_Postlaunch          = SNNN.NNN     /* TBS by SBRS */
      B1L_RTemp_Current             = SNNN.NNN     /* TBS by SBRS */
      B2L_RTemp_Prelaunch           = SNNN.NNN     /* TBS by SBRS */
      B2L_RTemp_Postlaunch          = SNNN.NNN     /* TBS by SBRS */
      B2L_RTemp_Current             = SNNN.NNN     /* TBS by SBRS */
      B3L_RTemp_Prelaunch           = SNNN.NNN     /* TBS by SBRS */
      B3L_RTemp_Postlaunch          = SNNN.NNN     /* TBS by SBRS */
      B3L_RTemp_Current             = SNNN.NNN     /* TBS by SBRS */
      B4L_RTemp_Prelaunch           = SNNN.NNN     /* TBS by SBRS */
      B4L_RTemp_Postlaunch          = SNNN.NNN     /* TBS by SBRS */
      B4L_RTemp_Current             = SNNN.NNN     /* TBS by SBRS */
      B5L_RTemp_Prelaunch           = SNNN.NNN     /* TBS by SBRS */
      B5L_RTemp_Postlaunch          = SNNN.NNN     /* TBS by SBRS */
      B5L_RTemp_Current             = SNNN.NNN     /* TBS by SBRS */
```

```
      B6L_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B6L_RTemp_Postlaunch           = SNNN.NNN     /* TBS by SBRS */
      B6L_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
      B7L_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B7L_RTemp_Postlaunch           = SNNN.NNN     /* TBS by SBRS */
      B7L_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
      B8L_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B8L_RTemp_Postlaunch           = SNNN.NNN     /* TBS by SBRS */
      B8L_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
    END_GROUP = REFERENCE_LOW
    GROUP = REFERENCE_HIGH
      B1H_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B1H_RTemp_Postlaunch           = SNNN.NNN     /* TBS by SBRS */
      B1H_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
      B2H_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B2H_RTemp_Postlaunch    = SNNN.NNN     /* TBS by SBRS */
      B2H_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
      B3H_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B3H_RTemp_Postlaunch    = SNNN.NNN     /* TBS by SBRS */
      B3H_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
      B4H_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B4H_RTemp_Postlaunch    = SNNN.NNN     /* TBS by SBRS */
      B4H_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
      B5H_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B5H_RTemp_Postlaunch    = SNNN.NNN     /* TBS by SBRS */
      B5H_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
      B6H_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B6H_RTemp_Postlaunch    = SNNN.NNN     /* TBS by SBRS */
      B6H_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
      B7H_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B7H_RTemp_Postlaunch    = SNNN.NNN     /* TBS by SBRS */
      B7H_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
      B8H_RTemp_Prelaunch            = SNNN.NNN     /* TBS by SBRS */
      B8H_RTemp_Postlaunch    = SNNN.NNN     /* TBS by SBRS */
      B8H_RTemp_Current              = SNNN.NNN     /* TBS by SBRS */
    END_GROUP = REFERENCE_HIGH
END_GROUP = REFERENCE_TEMPERATURES

GROUP = SENSITIVITY_TEMPERATURES
    GROUP = SENSITIVITY_LOW
      B1L_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B1L_SCoeff_Postlaunch          = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B1L_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B2L_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B2L_SCoeff_Postlaunch          = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B2L_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B3L_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B3L_SCoeff_Postlaunch          = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B3L_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B4L_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B4L_SCoeff_Postlaunch          = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B4L_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B5L_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B5L_SCoeff_Postlaunch          = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B5L_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6L_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6L_SCoeff_Postlaunch          = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6L_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6L_SCoeffOff_Prelaunch = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6L_SCoeffOff_Postlaunch       = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6L_SCoeffOff_Current  = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B7L_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B7L_SCoeff_Postlaunch          = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B7L_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B8L_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B8L_SCoeff_Postlaunch          = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B8L_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
    END_GROUP = SENSITIVITY_LOW
    GROUP = SENSITIVITY_HIGH

      B1H_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B1H_SCoeff_Postlaunch          = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B1H_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
```

```
      B2H_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B2H_SCoeff_Postlaunch  = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B2H_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B3H_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B3H_SCoeff_Postlaunch  = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B3H_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B4H_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B4H_SCoeff_Postlaunch  = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B4H_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B5H_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B5H_SCoeff_Postlaunch  = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B5H_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6H_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6H_SCoeff_Postlaunch  = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6H_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6H_SCoeffOff_Prelaunch = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6H_SCoeffOff_Postlaunch       = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B6H_SCoeffOff_Current  = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B7H_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B7H_SCoeff_Postlaunch  = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B7H_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B8H_SCoeff_Prelaunch           = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B8H_SCoeff_Postlaunch  = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
      B8H_SCoeff_Current             = (SNNN.NNNN, SNNN.NNNN,....) /* TBS by SBRS */
    END_GROUP = SENSITIVITY_HIGH
END_GROUP = SENSITIVITY_TEMPERATURES

GROUP = LAMP_RADIANCE
    GROUP = TRENDING_COEFFS
      Lamp1_Coeffs    = (SNNN.NNNNNNN, SNNN.NNNNNNN)       /* 2 values TBS by LPSO */
      Lamp2_Coeffs    = (SNNN.NNNNNNN, SNNN.NNNNNNN)       /* 2 values TBS by LPSO */
    END_GROUP = TRENDING_COEFFS
    GROUP = LAMP_RADIANCE_LOW
      B1L_Rad_State1_Prelaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1L_Rad_State1_Postlaunch            = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1L_Rad_State1_Current        = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B1L_Rad_State2_Prelaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1L_Rad_State2_Postlaunch            = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1L_Rad_State2_Current        = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B1L_Rad_State3_Prelaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1L_Rad_State3_Postlaunch            = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1L_Rad_State3_Current        = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B2L_Rad_State1_Prelaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2L_Rad_State1_Postlaunch            = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2L_Rad_State1_Current        = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B2L_Rad_State2_Prelaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2L_Rad_State2_Postlaunch            = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2L_Rad_State2_Current        = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B2L_Rad_State3_Prelaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2L_Rad_State3_Postlaunch            = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2L_Rad_State3_Current        = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B3L_Rad_State1_Prelaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3L_Rad_State1_Postlaunch            = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3L_Rad_State1_Current        = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B3L_Rad_State2_Prelaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3L_Rad_State2_Postlaunch            = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3L_Rad_State2_Current        = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
```

```
      B3L_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3L_Rad_State3_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3L_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B4L_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4L_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4L_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B4L_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4L_Rad_State2_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4L_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B4L_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4L_Rad_State3_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4L_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B5L_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5L_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5L_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B5L_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5L_Rad_State2_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5L_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B5L_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5L_Rad_State3_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5L_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B7L_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7L_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7L_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B7L_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7L_Rad_State2_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7L_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B7L_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7L_Rad_State3_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7L_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B8L_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8L_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8L_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by LPSO */
      B8L_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8L_Rad_State2_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8L_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by LPSO */
      B8L_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8L_Rad_State3_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8L_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by LPSO */
    END_GROUP = LAMP_RADIANCE_LOW
    GROUP = LAMP_RADIANCE_HIGH
      B1H_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1H_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1H_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
```

```
      B1H_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1H_Rad_State2_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1H_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B1H_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1H_Rad_State3_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B1H_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B2H_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2H_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2H_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B2H_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2H_Rad_State2_Postlaunch       = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B2H_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B2H_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B2H_Rad_State3_Postlaunch       = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B2H_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B3H_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3H_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3H_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B3H_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3H_Rad_State2_Postlaunch       = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B3H_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B3H_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B3H_Rad_State3_Postlaunch       = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B3H_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B4H_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4H_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4H_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B4H_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4H_Rad_State2_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4H_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B4H_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B4H_Rad_State3_Postlaunch       = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B4H_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B5H_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5H_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5H_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B5H_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5H_Rad_State2_Postlaunch       = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B5H_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B5H_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B5H_Rad_State3_Postlaunch       = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B5H_Rad_State3_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B7H_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7H_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7H_Rad_State1_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B7H_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7H_Rad_State2_Postlaunch       = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B7H_Rad_State2_Current          = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
```

```
      B7H_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by
LPSO */
      B7H_Rad_State3_Postlaunch      = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B7H_Rad_State3_Current         = (NNN.NNNN, NNN.NNNN,...)    /* 16 values TBS by LPSO */
      B8H_Rad_State1_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8H_Rad_State1_Postlaunch             = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8H_Rad_State1_Current         = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by LPSO */
      B8H_Rad_State2_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8H_Rad_State2_Postlaunch      = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by LPSO */
      B8H_Rad_State2_Current         = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by LPSO */
      B8H_Rad_State3_Prelaunch              = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by
LPSO */
      B8H_Rad_State3_Postlaunch      = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by LPSO */
      B8H_Rad_State3_Current         = (NNN.NNNN, NNN.NNNN,...)    /* 32 values TBS by LPSO */
    END_GROUP = LAMP_RADIANCE_HIGH
END_GROUP = LAMP_RADIANCE

GROUP = LAMP_REFERENCE
    Lmp_Rtemp_PreLaunch                     = (SNNN.NNNN, SNNN.NNNN,......)      /* 14 values
TBS by LPSO */
    Lmp_Rtemp_Postlaunch                    = (SNNN.NNNN, SNNN.NNNN,......)      /* 14 values
TBS by LPSO */
    Lmp_Rtemp_Current               = (SNNN.NNNN, SNNN.NNNN,......)      /* 14 values TBS by
LPSO */
END_GROUP = LAMP_REFERENCE

GROUP = REFLECTIVE_IC_COEFFS
    GROUP = REFLECT_IC_COEFFS_LOW
      B1L_Coefficients_Detector1            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector2            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector3            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector4            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector5            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector6            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector7            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector8            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector9            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector10           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector11           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector12           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector13           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector14           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector15           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1L_Coefficients_Detector16           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector1            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector2            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector3            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector4            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector5            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
```

```
      B2L_Coefficients_Detector6          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector7          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector8          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector9          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector10         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector11         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector12         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector13         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector14         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector15         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2L_Coefficients_Detector16         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector1          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector2          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector3          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector4          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector5          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector6          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector7          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector8          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector9          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector10         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector11         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector12         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector13         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector14         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector15         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3L_Coefficients_Detector16         = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector1          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector2          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector3          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector4          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector5          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector6          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector7          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector8          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector9          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
```

```
      B4L_Coefficients_Detector10          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector11          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector12          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector13          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector14          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector15          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4L_Coefficients_Detector16          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO *
      B5L_Coefficients_Detector1           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector2           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector3           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector4           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector5           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector6           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector7           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector8           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector9           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector10          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector11          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector12          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector13          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector14          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector15          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5L_Coefficients_Detector16          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector1           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector2           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector3           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector4           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector5           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector6           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector7           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector8           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector9           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector10          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector11          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector12          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector13          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
```

```
      B7L_Coefficients_Detector14           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector15           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7L_Coefficients_Detector16           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector1            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector2            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */¤
      B8L_Coefficients_Detector3            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector4            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector5            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector6            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector7            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector8            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector9            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector10           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector11           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector12           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector13           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector14           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector15           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector16           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector17           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector18           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector19           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector20    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector21           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector22    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector23    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector24    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector25    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector26    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector27    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector28    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector29    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector30    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8L_Coefficients_Detector31           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8L_Coefficients_Detector32    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
    END_GROUP = REFLECT_IC_COEFFS_LOW
    GROUP = REFLECT_IC_COEFFS_HIGH
```

```
      B1H_Coefficients_Detector1           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector2           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector3           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector4           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector5           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector6           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector7           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector8           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector9           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector10          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector11          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector12          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector13          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector14          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector15          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B1H_Coefficients_Detector16          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector1           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector2           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector3           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector4           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector5           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector6           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector7           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector8           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector9           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector10    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B2H_Coefficients_Detector11          = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B2H_Coefficients_Detector12    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B2H_Coefficients_Detector13    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B2H_Coefficients_Detector14    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B2H_Coefficients_Detector15    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B2H_Coefficients_Detector16    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B3H_Coefficients_Detector1           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3H_Coefficients_Detector2           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3H_Coefficients_Detector3           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3H_Coefficients_Detector4           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
```

```
      B3H_Coefficients_Detector5            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3H_Coefficients_Detector6            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3H_Coefficients_Detector7            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3H_Coefficients_Detector8            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3H_Coefficients_Detector9            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3H_Coefficients_Detector10    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B3H_Coefficients_Detector11           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B3H_Coefficients_Detector12    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B3H_Coefficients_Detector13    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B3H_Coefficients_Detector14    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B3H_Coefficients_Detector15    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B3H_Coefficients_Detector16    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B4H_Coefficients_Detector1            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector2            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector3            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector4            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector5            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector6            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector7            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector8            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector9            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector10    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B4H_Coefficients_Detector11           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B4H_Coefficients_Detector12    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B4H_Coefficients_Detector13    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B4H_Coefficients_Detector14    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B4H_Coefficients_Detector15    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B4H_Coefficients_Detector16    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B5H_Coefficients_Detector1            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5H_Coefficients_Detector2            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5H_Coefficients_Detector3            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5H_Coefficients_Detector4            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5H_Coefficients_Detector5            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5H_Coefficients_Detector6            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5H_Coefficients_Detector7            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5H_Coefficients_Detector8            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
```

```
      B5H_Coefficients_Detector9            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5H_Coefficients_Detector10    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B5H_Coefficients_Detector11           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B5H_Coefficients_Detector12    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B5H_Coefficients_Detector13    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B5H_Coefficients_Detector14    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B5H_Coefficients_Detector15    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B5H_Coefficients_Detector16    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B7H_Coefficients_Detector1            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector2            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector3            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector4            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector5            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector6            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector7            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector8            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector9            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector10    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B7H_Coefficients_Detector11           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B7H_Coefficients_Detector12    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B7H_Coefficients_Detector13    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B7H_Coefficients_Detector14    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B7H_Coefficients_Detector15    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B7H_Coefficients_Detector16    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8H_Coefficients_Detector1            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector2            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector3            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector4            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector5            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector6            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector7            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector8            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector9            = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector10    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
      B8H_Coefficients_Detector11           = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values
TBS by LPSO */
      B8H_Coefficients_Detector12    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)   /* 18 values TBS by
LPSO */
```

```
       B8H_Coefficients_Detector13    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector14    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector15    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector16    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector17    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector18    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector19    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector20    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector21    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector22    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector23    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector24    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector25    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector26    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector27    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector28    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector29    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector30    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector31    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
       B8H_Coefficients_Detector32    = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /* 18 values TBS by
LPSO */
     END_GROUP = REFLECT_IC_COEFFS_HIGH
END_GROUP = REFLECT_IC_COEFFS


GROUP = B6_VIEW_COEFFS
       B6_View_Coefficients_Detector1 = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /*  15 values TBS by
LPSO */
       B6_View_Coefficients_Detector2 = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /*  15 values TBS by
LPSO */
       B6_View_Coefficients_Detector3 = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /*  15 values TBS by
LPSO */
       B6_View_Coefficients_Detector4 = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /*  15 values TBS by
LPSO */
       B6_View_Coefficients_Detector5 = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /*  15 values TBS by
LPSO */
       B6_View_Coefficients_Detector6 = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /*  15 values TBS by
LPSO */
       B6_View_Coefficients_Detector7 = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /*  15 values TBS by
LPSO */
       B6_View_Coefficients_Detector8 = (SNNN.NNNNNNN, SNNN.NNNNNNN,...)    /*  15 values TBS by
LPSO */
END_GROUP = B6_VIEW_COEFFS

GROUP = B6_TEMP_MODEL_COEFFS
     B6_Temp_Model_Parm                      = (+1.0,+0.0,+0.0,+0.0,+0.0,+0.0)     /* 6 values
supplied by LPSO */
END_GROUP = B6_TEMP_MODEL_COEFFS

GROUP = THERMISTOR_COEFFS
     Black_Body_Isolated_Temp         = (NNN.NNNNNNN, NNN.NNNNNNN,...)      /* 6 values TBS by
LPSO */
```

```
     Black_Body_Control_Temp          = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Cold_FP_Control_Temp                   = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Cold_FP_Monitor_Temp                   = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Cal_Shutter_Flag_Temp                  = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Backup_Shutter_Flag_Temp         = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Baffle_Heater_Temp                     = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Silicon_FP_Array_Temp                  = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Primary_Mirror_Temp                    = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Secondary_Mirror_Temp                  = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Scan_Line_Corrector_Temp         = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Baffle3_Tube_Temp                = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Baffle2_Support_Temp                   = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Cal_Lamp_Housing_Temp            = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Cal_Shutter_Hub_Temp                   = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Ambient_Preamp_HighCh_Temp       = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Ambient_Preamp_LowCh_Temp = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by LPSO */
     Postamp_Temp_B4                  = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Cold_Preamp_B7_Temp                    = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Pan_Band_Postamp_Temp            = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Telescope_Housing_Temp           = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Primary_Mirror_Mask_Temp         = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Secondary_Mirror_Mask_Temp       = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Telescope_Baseplate_Temp               = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Mux1_Power_Supply_Temp           = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Mux1_Electronics_Temp                  = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
     Mux2_Power_Supply_Temp           = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS by
LPSO */
     Mux2_Electronics_Temp                  = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 6 values TBS
by LPSO */
END_GROUP = THERMISTOR_COEFFS

GROUP = LAMP_CURRENTS
     tec_lamp_i1                      = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 2 values TBS by
SBRS */
     tec_lamp_i2                      = (NNN.NNNNNNN, NNN.NNNNNNN,...)     /* 2 values TBS by
SBRS */
END_GROUP = LAMP_CURRENTS


GROUP = FILL_PATTERNS
     Band_Fill_Pattern = (0, 255)
END_GROUP_FILL_PATTERNS


END
```

Acronyms and Abbreviations

0R	Level 0 reformatted (data)
ACCA	automated cloud cover assessment
ADS	angular displacement sensor data
ADSA	attitude displacement sensor assembly
ASCII	American Standard Code for Information Interchange
CCB	Configuration Control Board
CPF	calibration parameter file
CCSDS	Consultative Committee for Space Data Systems
DAAC	Distributed Active Archive Center
DN	digital number
ECS	EOSDIS Core System
EDC	EROS Data Center
EOL	end of line
EOS	Earth Observing System
EOSDIS	EOS Data and Information System
EROS	Earth Resources Observation Systems
ETM+	Enhanced Thematic Mapper Plus
EU	engineering unit
FASC	full aperture solar calibrator
HDF	Hierarchical Data Format
I/O	input/output
IAS	Image Assessment System
IC	internal calibrator
IEEE	Institute of Electrical and Electronics Engineers
IFOV	instrument field of view
IMU	inertial measurement unit
ISO	International Organization for Standardization
JPL	Jet Propulsion Laboratory

kHz	kilohertz
km	kilometer
L1R	Level 1 radiometrically corrected (data)
LMC	Lockheed Martin Corporation
LPS	Landsat Processing System
LPSO	Landsat Project Science Office
Mbps	megabit per second
MTFC	modulation transfer function compensation
NEOS	National Earth Orientation Service
NSDI	normalized snow difference index
ODL	Object Description Language
PCD	payload correction data
PVL	Parameter Value Language
SAM	scan angle monitor
SBRS	Santa Barbara Remote Sensing
SME	scan mirror electronics
TBR	to be resolved
TBS	to be submitted
URL	Uniform Resource Locator
UTC	Universal Time Code
UT1	UTC corrected
WRS	Worldwide Reference System