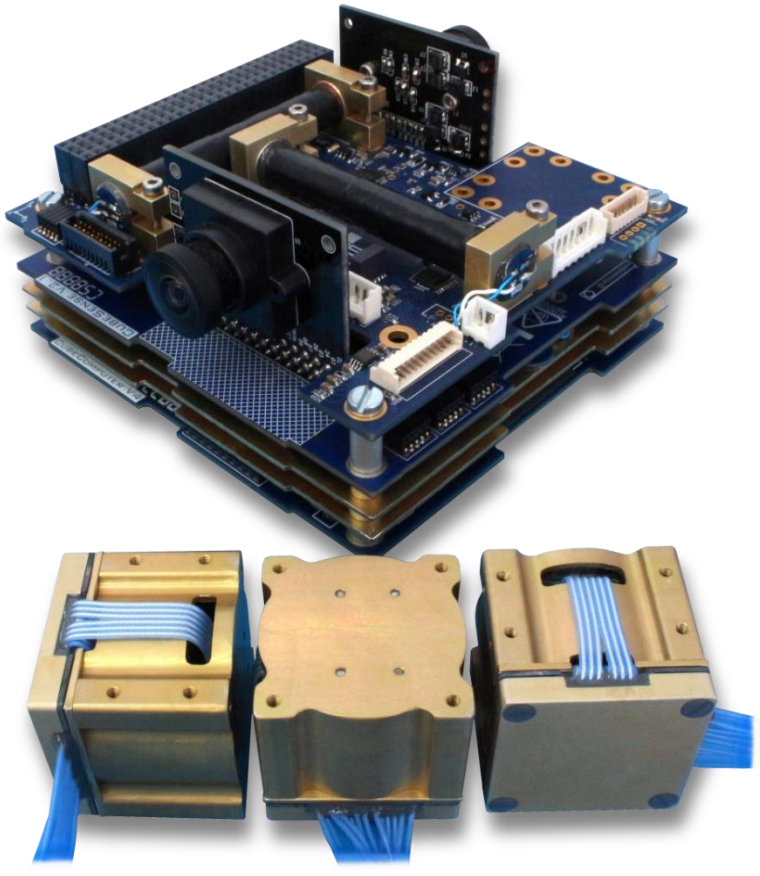
CubeADCS

The complete ADCS solution



Firmware Reference Manual

Contact Us

Phone (0027) 79 945 9957
E-mail info@cubespace.co.za
Web www.cubespace.co.za
Facebook /CubeSpaceADCS
Twitter @CubeSpace_ADCS

Physical Address

CubeSpace
Hammanshand Road
Stellenbosch
7600
South Africa

Document	CubeADCS Firmware Reference Manual
Version	7.4
Domain	Public
Date modified	January 2022
Approved by	Name: Lourens Visagie
	Signature:

Table of Contents

List of Acronyms/Abbreviations

ACP	ADCS Control Program
ADCS	Attitude Determination and Control System
CSS	Coarse Sun Sensor
ESD	Electrostatic Discharge
I^2C	Inter-Integrated Circuit
MCU	Microcontroller Unit
MEMS	Microelectromechanical System
OBC	Onboard Computer
PCB	Printed Circuit Board
RTC	Real-Time Clock
SBC	Satellite Body Coordinate
SPI	Serial Peripheral Interface
TC	Telecommand
TLM	Telemetry
UART	Universal Asynchronous Receiver/Transmitter

Relevant reference documents

This document is to be used in combination with the following documents:

Reference	Document name	Document version
Ref 1	CubeADCS – ICD	V3.18 or higher
Ref 2	CubeADCS – User Manual	V3.09 or higher

Relevant Firmware Versions

This document serves as reference to the CubeACP and bootloader with the following version numbers:

	CubeACP	Bootloader
Software version	7.x	3.x
Node type identifier	10	110
Interface version	7	3

1. Introduction

This document contains the necessary information to interface with the CubeSpace bootloader and CubeACP applications.

The bootloader operation, application selection, and interaction through the bootloader are detailed in this manual.

It further describes the protocol required to communicate with the CubeComputer, running either the bootloader or CubeACP application, using I^2C, UART and CAN communication channels. It also details all the low-level telecommands and telemetry frames that can be sent or requested.

This manual also provides required procedures (sequences of commands and telemetry requests) to upload new firmware to the CubeComputer external flash memory or to download files from the CubeComputer SD card.

2. CubeSpace Firmware Programs

2.1 Bootloader

The CubeSpace CubeComputer makes use of a bootloader when it is used in a CubeADCS unit. All CubeComputer units delivered by CubeSpace have the bootloader installed. The bootloader performs several important tasks required for the CubeSpace CubeComputer infrastructure, such as

- Initialises external SRAM so that applications can make use of it
- Enables internal and external watchdogs of the MCU and prevents faulty or damaged applications from halting the CPU.
- Allows reprogramming of application via UART, I^2C, or CAN interface
- Counts number of reboots and tracks boot attempts
- Provides status telemetry

The bootloader is accessible via UART, I^2C, and CAN.

2.2 ADCS Control Program (ACP)

The ACP (or CubeACP) is the application that runs on an integrated CubeADCS unit and performs the attitude control functions. This manual lists the detail of all the low-level messages that can be sent to or received from the ACP, however it should be read in combination with the CubeADCS User Manual [Ref 2], which contains the functional description of the ACP.

CubeComputer units in CubeADCS bundles come pre-loaded with the bootloader and the ACP as the default application.

2.3 Bootloader to Control Program Transition

The bootloader to control program sequence is explained by the flow diagram in Figure 1.

The bootloader will start to execute when the CubeComputer is powered up or reset. The bootloader will run for **5 seconds** before it will attempt to boot the application.

During the 5 seconds that the bootloader is active, it is possible to halt the boot process, so that further status telemetry can be obtained from the bootloader, or to reprogram the application that should run after the bootloader.

The boot process is stopped and the bootloader will remain running when an identification telemetry request (see Table 27: Identification Telemetry Format) is sent to the bootloader. If the bootloader is not interrupted in the first 5s, it will continue to attempt to run the ADCS control program (ACP).

The Boot Status is an enumeration value that is stored in the CubeComputer flash memory and keeps track of the previous boot attempt.

If, for some reason, the application failed to start, and the bootloader executes again, it will attempt to boot the application only two more times before changing the Boot Status value to "3x failed attempts". At this point the bootloader will no longer reboot the application and remain active indefinitely, or until the Boot Status is reset.

The Boot Status can be queried using a *Boot Index and Status* telemetry request (see Table 32: Boot Index and Status Telemetry Format). This status can use used to determine if an error condition exists on the CubeComputer.

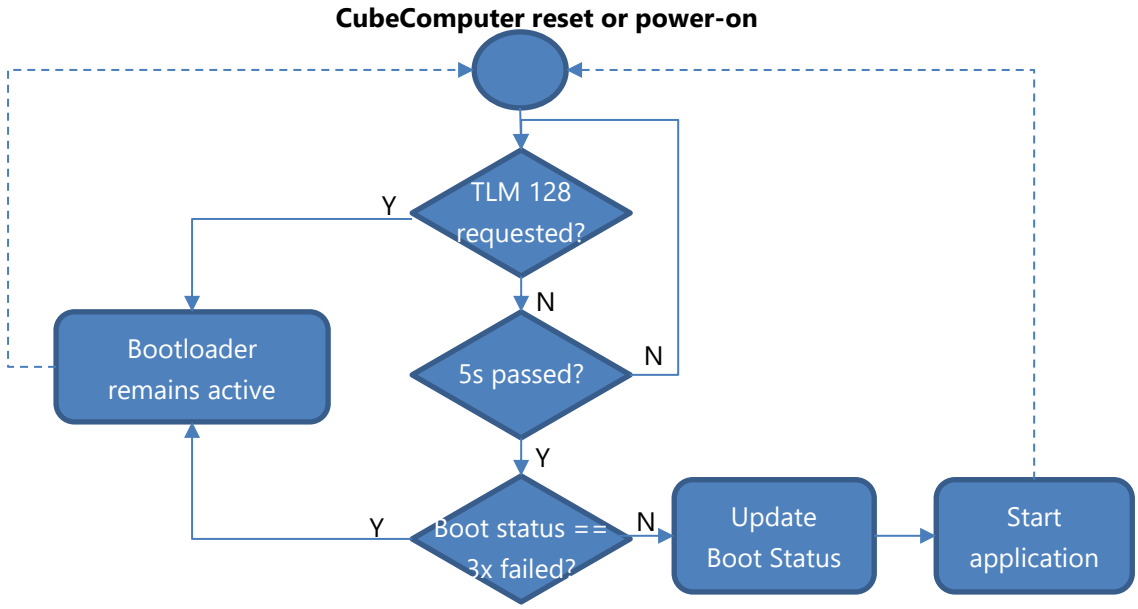


Figure 1: Bootloader flow diagram

The possible values for the Boot Status are listed in Table 1.

Table 1: BootStatus Enumeration Values

Numeric Value	Name	Description
0	New Selection	New Selection
1	Boot Success	Boot Success
2	1 Failed boot attempt	1 Failed boot attempt
3	2 Failed boot attempts	2 Failed boot attempts
4	3 Failed boot attempts	3 Failed boot attempts

It is also possible to reset the *Boot Status* to *New Selection*, by sending a *Set Boot Index* telecommand (see Table 62: Set Boot Index Command Format). Note that the *Program Index*

parameter of this command should always be set to *Internal Flash Program*. Other options are not supported.

A *Reset Counter* is also stored in flash memory and maintained by the bootloader. The *Reset Counter* is incremented every time the bootloader starts, and can be queried through the *Boot and Running Program Status* telemetry request (see Table 28: Boot And Running Program Status Telemetry Format). This latter telemetry frame also provides the cause of the last MCU reset, read from a hardware register on the MCU. The *Boot Cause* shows the reason why the CubeComputer was reset.

The current state of the bootloader can be queried through the *Bootloader State* telemetry frame (see Table 69: Bootloader State Telemetry Format). This telemetry frame shows for how long the bootloader has been running, SRAM enabled state and error flags.

The error flags in the above *Bootloader State* telemetry frame are latched. They remain set until cleared using the Clear Error Flags command (see Table 61: Clear Error Flags Command Format). This command has no parameters.

It is possible to force the bootloader to start the application program by sending a Run Selected Program command (see Table 64: Run Selected Program Command Format).

The *Reset Boot Registers* command (see Table 13: Reset Boot Registers Command Format) will reset the Boot Counter, Boot State and Boot Cause registers. It has no parameters.

2.4 Multiple Application Support

It is possible to store up to 7 ACP programs in the CubeComputer's external flash memory. Each program area has a maximum size of 512kB.

The ACP program instance that should be started by the bootloader is selected by copying it to the internal MCU flash. This action is performed by sending a *Copy Program to Internal Flash* telecommand (Table 67: Copy Program to Internal Flash Command Format) to the bootloader with the appropriate parameters.

For further details on operational sequences, such as uploading program, please see section 6.

3. Operations Common to the Bootloader and the CubeACP Application

Both the Bootloader and the ACP support the Identification telemetry request (Table 27: Identification Telemetry Format), although the different programs will reply with different information which allows for the interfacing software to identify which program is currently running.

Most of the status telemetry frames mentioned in the previous bootloader section can also be requested while the ACP is running. This includes the *Boot and Running Program Status*, and *Boot Index and Status* telemetry frames.

The *SRAM Latchup counters* telemetry frame (Table 35: SRAM Latchup counters Telemetry Format) gives information on the number of SRAM latch-up events that have occurred, and the *EDAC Error Counters* telemetry frame (Table 36: EDAC Error Counters Telemetry Format) provides the number of EDAC errors that have been detected and corrected while scrubbing the SRAM.

The *Communication Status* telemetry frame (Table 37: Communication Status Telemetry Format) returns the number of commands and telemetry requests that have been served, as well as error flags related to use of the communication channel.

The *Telecommand Acknowledge* telemetry frame (Table 39: Telecommand Acknowledge Telemetry Format) can be requested to check the status of the last telecommand that was sent. This telemetry frame is intended to be used only with I^2C communication – see section 4.3.2 for details.

It is possible to reset the CubeComputer using the *Reset* telecommand (Table 10: Reset Command Format) while either the bootloader or the CubeACP is running.

The Current Unix Time can be requested or changed using the *Current Unix Time* get and set messages (Table 50: Current Unix Time Message Format). It is also possible to periodically save the Unix time to flash memory, so that it will persist between power cycles. This behaviour can be adjusted using the *Unix Time Save to Flash* messages (Table 51: Unix Time Save to Flash Message Format).

Both the CubeACP program and the bootloader allows for downloading of files from the SD card, thus they both support the same set of telecommands and telemetries to allow for this process. See section 6 for details on this and other operational sequences that make use of sequences of commands and telemetry requests.

4. Formatting of Telecommands, Telemetry Requests and Replies

Communication to and from the CubeComputer utilises UART, I^2C, and CAN. This section contains the protocol details for each interface.

4.1 Telemetry and Telecommand IDs

A single byte message identifier is used for both telecommands and telemetry requests. The most significant bit of this message ID determines whether it is a telecommand or telemetry request, and the lower 7 bits contain the ID.

Table 2: Telecommand or telemetry frame ID

Bit(s)	Data
7	0 = telecommand, 1 = telemetry request
0:6	Telecommand or telemetry frame ID

When considering the full byte identifier, telecommands will have values in the range 0-127 and telemetry requests in the range 128-255.

4.2 UART

The specifications of the UART interface are given in Table 3.

Table 3: Standard UART specifications for CubeComputer

Parameter	Value
Baud rate	115200
Data bits	8
Parity	None
Stop bits	1

The UART protocol makes use of start-of-message (SOM 0x7F) and end-of-message (EOM 0xFF) identifiers to mark the beginning and end of a transmission. An escape character (ESC 0x1F) precedes the SOM and EOM identifiers. When decoding a UART message, on reception of the escape character, the byte following the escape character has the following implications:

Table 4: UART message decoding

Byte received after ESC character	Meaning
0x7F	Start of message
0xFF	End of message

0x1F	Data byte 0x1F
other	Should not occur (error)

4.2.1 UART Telemetry Request and Reply Format

A telemetry request via the UART will have the following form:

Figure 2: UART Telemetry request

The reply from the ADCS will then have the following form:

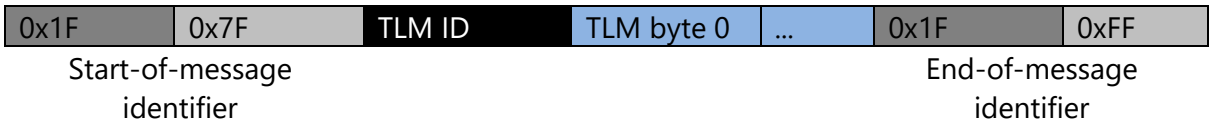


Figure 3: UART Telemetry reply

4.2.2 UART Telecommand Format

A telecommand to the ADCS, via the UART, will have the following form:

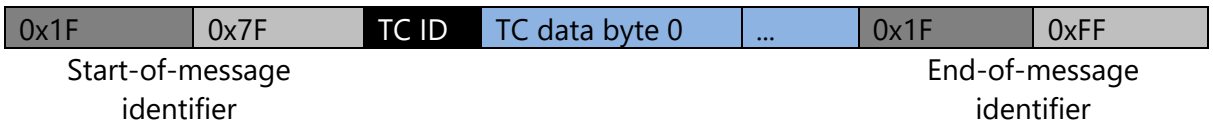


Figure 4: UART Telecommand

An acknowledge message will be sent as reply to the telecommand.

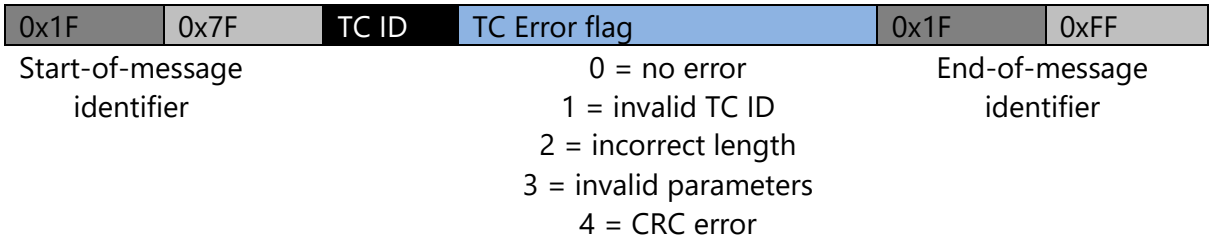


Figure 5: UART Telecommand acknowledge

The reply will contain two data bytes, the last one being the *TC Error* flag. The receipt of the telecommand acknowledge will indicate that another telecommand may be sent. Sending another telecommand before the acknowledge will corrupt the telecommand buffer.

4.3 I²C

CubeComputer acts as a slave on the system I²C bus. The Computer has a default 7-bit I²C slave address of 0x57.

Table 5: CubeComputer (slave node) I²C addresses.

	8-bit Hex byte	7-bit Hex address	Binary
I²C write	0xAE	0x57	0b0010 010**0**
I²C read	0xAF	0x57	0b0010 010**1**

4.3.1 I²C Telemetry Request and Reply Format

Telemetry is requested from the ADCS over the system I²C bus by either performing a combined read-write operation (repeated start condition) or a separate master write to select the TLM register, followed by a master read operation. The first write following the start condition is the address of the node (0xAE). This is followed by the telemetry frame identifier. In the case of a separate write and read, a stop condition will follow, and the master will then issue another start condition. In the case of the combined write-then-read operation, the master will issue a repeated start condition (without a preceding stop condition).

The second start condition will be followed by the node read address (0xAF). The master then issues several read cycles depending on the length of the telemetry frame.

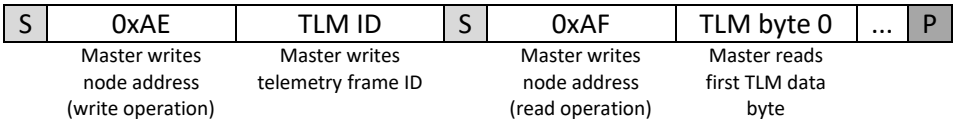


Figure 6: I2C Telemetry request using I2C repeated start condition

Because the master determines the number of bytes that are read, it is possible to read past the end of a telemetry frame or to read an incomplete telemetry frame. The ADCS will set an error flag if an incorrect number of bytes are read for a given TLM ID. This flag is stored in the *Communication Status* frame and can be read using a telemetry request. The flag will remain set until the *Clear Errors* telecommand is issued.

4.3.2 I²C Telecommand Format

Telecommands are sent by performing a master write to the module. The first data byte (after the address byte) is the telecommand identifier, followed by the telecommand parameters.

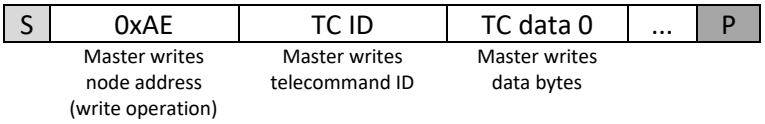


Figure 7: I2C Telecommand

Because the ADCS is an I²C slave, it cannot acknowledge telecommands by performing an I²C write transaction. The telecommand acknowledge status must therefore be polled via a *Telecommand Acknowledge* telemetry request.

It is not a requirement that the telecommand acknowledge status must be read following a telecommand, but an error will occur if another telecommand is sent before the Telecommand Processed flag (contained in the *Telecommand Acknowledge* frame) has been set. In this case the telecommand buffer will be overwritten while the first telecommand is being processed, leading to corrupt telecommand data.

The *Processed* flag is not an indication of the telecommand execution status. The *Processed* flag is only an indication that the module is ready for another telecommand to be sent.

The *Telecommand Acknowledge* frame also contains a *TC Error* flag. This flag will be set if an invalid telecommand ID was received for the last telecommand, or if the number of data bytes were incorrect or contained invalid data.

The following sequence illustrates the actions that the master must take to ensure proper telecommand execution:

1. Send telecommand.
2. Poll *Telecommand Acknowledge Telemetry Format* until the *Processed* flag equals 1.
3. Confirm telecommand validity by checking the *TC Error* flag of the last read *Telecommand Acknowledge Telemetry Format*.
4. Back to step 1 (if another telecommand is to be sent)

4.4 CAN

The CubeADCS acts as a slave on the CAN bus and supports 1 Mbps CAN bus V2.0 standard.

The CubeSpace CAN protocol uses an extended frame format, consisting of 29 identifier bits. The 29-bit frame identifier has the structure depicted in Table 6.

Table 6: CAN identifier frame format.

MSB			LSB
Message Type	**Channel/ID**	**Source Address**	**Destination Address**
5 Bits	8 Bits	8 Bits	8 Bits

The ADCS CubeComputer uses an 8-bit address of 0x01. Thus, all messages that are sent to the ADCS will have a value of 0x01 for the *Destination Address* field. Likewise, all replies coming from the ADCS will have a value of 0x01 in the *Source Address* field.

The *Message Type* field in the 29-bit CAN identifier can take on values as listed in.

Table 7: CAN message types.

Message Type	Identifier	Description
TC Request	0x01	Command/Request

TC Response	0x02	Acknowledge
TC Not Ack	0x03	Command Failure
TLM Request	0x04	Request
TLM Response	0x05	Response
TLM Not Ack	0x06	Request Failure
Extended TC Request	0x07	Multiple CAN Packet Command/Request
Extended TLM Response	0x08	

4.4.1 CAN Telemetry Request and Reply Format

A telemetry request to the ADCS, over CAN, will have the following contents for the frame identifier, data length code (DLC) and data field:

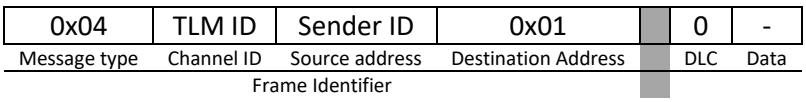


Figure 8: CAN Telemetry Request

The reply from the ADCS will then have the following contents:

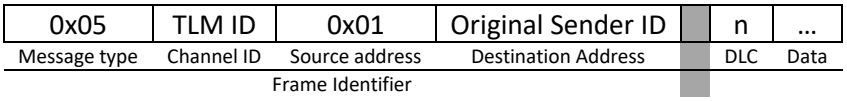


Figure 9: CAN Telemetry Reply

If an invalid telemetry ID was requested, the reply from the ADCS will have a message type of 0x06.

4.4.2 CAN Telecommand Format

A telecommand to the ADCS, over the CAN bus, will have the following contents in the frame identifier, DLC and data fields.

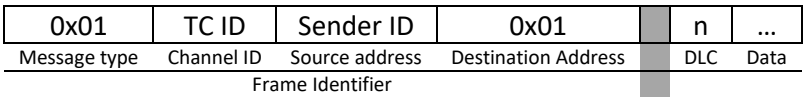


Figure 10: CAN Telecommand

The command will either be met with an acknowledge message (message ID of 0x02), or NACK message (message ID of 0x03).

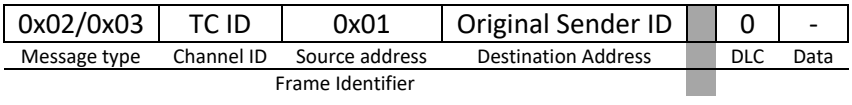


Figure 11: CAN Telecommand reply

4.4.3 Multi-message CAN Transactions

If the telecommand data or the telemetry response contains more than the maximum of eight data bytes, an Extended Telemetry Request or Extended Telecommand Response is used. The Message type for such TC or TLMs are defined in Table 7.

For these extended transactions, a protocol is implemented on the data bytes of the CAN packets. The 8th byte of the data of the extended transaction contains the number of CAN packets left which is part of the complete extended telemetry or telecommand request or response. Each packet will therefore only have 7 data bytes. Considering that the extended TLM or TC can consist of 256 (1 byte) such packets, the total length of an extended TC or TLM is 1792 bytes. If a TC or TLM is longer than 1792 bytes, the file download mechanism must be used.

Table 8: CAN Extended TC/TLM Data

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Packets Left

The last byte of the last packet must be zero indicating that there are no further CAN packets part of this transaction. A standard telecommand response, like the standard operations, is sent from the slave if the entire extended transaction has been received successfully.

4.5 CRC For Commands and Telemetry (CubeACP only)

When sending a command to the CubeACP, it is possible to include an 8-bit CRC checksum.

For instance, when sending a command that has a length of 8 bytes, it is possible to include a 9th byte that is computed from the previous 8 bytes. The extra byte will be interpreted as a checksum and used to validate the message. If the checksum fails, the command will be ignored. In the case of UART communications, the telecommand reply will have a value of 4 for the TC Error Flag (Figure 5). For I2C communication, the *Tc Error Status* in the *Telecommand Acknowledge* telemetry frame (Table 39: Telecommand Acknowledge Telemetry Format) will have a value of 4.

When requesting telemetry through I^2C, it is possible to read one extra byte past the allowed length of the telemetry frame. In this case, the extra byte will also be an 8-bit checksum computed by the CubeACP and can be used by the interfacing OBC to validate the message.

The C source code that is used on the CubeACP to compute the checksums is given below.

```
#define CRC_POLY 0x91
uint8_t CRC8Table[256];
// initialisation
```

```
// init lookup table for 8-bit crc calculation
void COMMS_Crc8Init()
{
    for (int I = 0; I < 256; i++)
    {
        val = I;
        for (int j = 0; j < 8; j++)
        {
            if (val & 1)
                val ^= CRC_POLY;
            val >>= 1;
        }
        CRC8Table[i] = val;
    }
}

/**************************************************************************//**
 * Calculates an 8-bit CRC value
 *
 * @param[in] buffer
 *   the buffer containing data for which to calculate the crc value
 * @param[in] len
 *   the number of bytes of valid data in the buffer
 *****************************************************************************/
uint8_t COMMS_Crc8Checksum(uint8_t* buffer, uint16_t len)
{
    if (len == 0) return 0xff;

    uint16_t I;
    uint8_t crc = 0;

    for (I = 0; I < len; i++)
        crc = CRC8Table[crc ^ buffer[i]];

    return crc;
}
```

5. Telecommand and Telemetry Message Definitions

This section lists all the telecommands and telemetry message formats that can be sent to the CubeACP or bootloader. The first sub-section lists messages that are common to both the bootloader and CubeACP, while following sections list the messages unique to the individual software programs.

The communications messages have further been grouped by telecommand (messages that can only be sent), telemetry frames (can only be requested) or configuration settings messages. The latter group of messages have matching get (telemetry) and set (telecommand) message formats.

5.1 Messages Common to the Bootloader and CubeACP

5.1.1 Common Telecommands

Table 9: List of Common Telecommands

ID	Name	Description	Length (bytes)
General			
1	Reset	Perform a reset – Table 10: Reset Command Format	1
4	Reset Log Pointer	Reset pointer to log buffer (from where LastLogEvent TLM is returned) – Table 11: Reset Log Pointer Command Format	0
5	Advance Log Pointer	Advance pointer to log buffer (from where LastLogEvent TLM is returned) – Table 12: Advance Log Pointer Command Format	0
6	Reset Boot Registers	Reset Boot Registers – Table 13: Reset Boot Registers Command Format	0
33	Format SD card	Format SD card – Table 14: Format SD card Command Format	1
108	Erase File	Erase File – Table 15: Erase File Command Format	3
112	Load File Download Block	Fill download buffer with file contents – Table 17: Load File Download Block Command Format	8
113	Advance File List Read Pointer	Advance File List Read Pointer – Table 18: Advance File List Read Pointer Command Format	0
114	Initiate File Upload	Initiate File Upload – Table 19: Initiate File Upload Command Format	2

115	File Upload Packet	File Upload Packet – Table 21: File Upload Packet Command Format	22
116	Finalize Upload Block	Finalize Uploaded File Block – Table 22: Finalize Upload Block Command Format	7
117	Reset Upload Block	Reset HoleMap for Upload Block – Table 23: Reset Upload Block Command Format	0
118	Reset File List Read Pointer	Reset File List Read Pointer – Table 24: Reset File List Read Pointer Command Format	0
119	Initiate Download Burst	Initiate Download Burst – Table 25: Initiate Download Burst Command Format	2

Table 10: Reset Command Format

ID	1		**Parameters Length (bytes)**	1	
Description	Perform a reset				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Magic number	UINT	Magic number to make sure it is a valid reset command. Should equal 0x5A

Table 11: Reset Log Pointer Command Format

ID	4	**Parameters Length (bytes)**	0
Description	Reset pointer to log buffer (from where LastLogEvent TLM is returned)		

Table 12: Advance Log Pointer Command Format

ID	5	**Parameters Length (bytes)**	0
Description	Advance pointer to log buffer (from where LastLogEvent TLM is returned)		

Table 13: Reset Boot Registers Command Format

ID	6	**Parameters Length (bytes)**	0
Description	Reset Boot Registers		

Table 14: Format SD card Command Format

ID	33	**Parameters Length (bytes)**	1

Description	Format SD card				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	Magic Number	UINT	Magic number to prevent against accidental format. Has to be set to 90 (0x5A)

Table 15: Erase File Command Format

ID	108		Parameters Length (bytes)		3
Description	Erase File				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	File Type	ENUM	File Type. Possible values are in Table 16: FileType Enumeration Values
	8	8	File Counter	UINT	File Counter
	16	1	Erase All	BOOL	Erase All

Table 16: FileType Enumeration Values

Numeric Value	Name	Description
2	Telemetry Log	Telemetry Log File
3	JPG Image	JPG Image File
4	BMP Image	BMP Image File
15	Index	Index File

Table 17: Load File Download Block Command Format

ID	112		Parameters Length (bytes)		8
Description	Fill download buffer with file contents				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	File Type	ENUM	File Type. Possible values are in Table 16: FileType Enumeration Values
	8	8	Counter	UINT	Counter
	16	32	Offset	UINT	Offset
	48	16	BlockLength	UINT	Block Length

Table 18: Advance File List Read Pointer Command Format

ID	113	**Parameters Length (bytes)**	0
Description	Advance File List Read Pointer		

Table 19: Initiate File Upload Command Format

ID	114		**Parameters Length (bytes)**		2
Description	Initiate File Upload				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Destination	ENUM	Destination. Possible values are in Table 20: FileUploadDestination Enumeration Values
	8	8	BlockSize	UINT	Block Size

Table 20: FileUploadDestination Enumeration Values

Numeric Value	Name	Description
2	EEPROM	EEPROM
3	Flash program 1	Flash program 1
4	Flash program 2	Flash program 2
5	Flash program 3	Flash program 3
6	Flash program 4	Flash program 4
7	Flash program 5	Flash program 5
8	Flash program 6	Flash program 6
9	Flash program 7	Flash program 7
10	SD User file 1	SD User file 1
11	SD User file 2	SD User file 2
12	SD User file 3	SD User file 3
13	SD User file 4	SD User file 4
14	SD User file 5	SD User file 5
15	SD User file 6	SD User file 6
16	SD User file 7	SD User file 7
17	SD User file 8	SD User file 8

Table 21: File Upload Packet Command Format

ID	115	**Parameters Length (bytes)**	22
Description	File Upload Packet		

Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Packet Number	UINT	Packet Number
	16	160	File Bytes	ARRAY	File Bytes

Table 22: Finalize Upload Block Command Format

ID	116		Parameters Length (bytes)		7
Description	Finalize Uploaded File Block				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	Destination	ENUM	Destination. Possible values are in Table 20: FileUploadDestination Enumeration Values
	8	32	Offset	UINT	Offset into file
	40	16	Block Length	UINT	Length of block

Table 23: Reset Upload Block Command Format

ID	117	Parameters Length (bytes)	0
Description	Reset HoleMap for Upload Block		

Table 24: Reset File List Read Pointer Command Format

ID	118	Parameters Length (bytes)	0
Description	Reset File List Read Pointer		

Table 25: Initiate Download Burst Command Format

ID	119		Parameters Length (bytes)		2
Description	Initiate Download Burst				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	Message Length	UINT	Message Length
	8	1	Ignore Hole Map	BOOL	Ignore Hole Map

5.1.2 Common Telemetry

Table 26: List of Common Telemetry Frames

ID	Name	Description	Length (bytes)
General			
128	Identification	Identification information for this node – Table 27: Identification Telemetry Format	8
129	Boot And Running Program Status	Boot And Running Program Status – Table 28: Boot And Running Program Status Telemetry Format	6
130	Boot Index and Status	Current selected boot index and status of last boot – Table 32: Boot Index and Status Telemetry Format	2
141	Last Logged Event	Last Logged Event (relative to pointer – adjusted via Advance and Reset TCs (3 & 4) – Table 34: Last Logged Event Telemetry Format	6
234	SD card format/erase progress	SD card format or erase progress – Table 38: SD card format/erase progress Telemetry Format	1
240	Telecommand Acknowledge	Telemetry frame with acknowledge status of the previously sent command – Table 39: Telecommand Acknowledge Telemetry Format	4
241	File Download Buffer with File Contents	File Download buffer 20-byte packet – Table 41: File Download Buffer with File Contents Telemetry Format	22
242	Download Block Ready	Status about download block preparation – Table 42: Download Block Ready Telemetry Format	5
243	File Information	File Information – Table 43: File Information Telemetry Format	12
244	Initialize Upload Complete	Initialize Upload Complete – Table 44: Initialize Upload Complete Telemetry Format	1
245	Upload Block Complete	Finalize Upload Block Complete – Table 45: Upload Block Complete Telemetry Format	1
246	Block Checksum	File upload Block CRC16 Checksum – Table 46: Block Checksum Telemetry Format	2
Satellite State			
142	SRAM Latchup counters	SRAM Latchup counters – Table 35: SRAM Latchup counters Telemetry Format	6
143	EDAC Error Counters	EDAC Error Counters – Table 36: EDAC Error Counters Telemetry Format	6
144	Communication Status	Communication status – includes command and telemetry counters and error flags – Table 37: Communication Status Telemetry Format	6

Table 27: Identification Telemetry Format

ID	128		Frame Length (bytes)		8
Description	Identification information for this node				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	Node type	UINT	Node type identifier.
	8	8	Interface version	UINT	Interface version. This field should have a value of 1
	16	8	Firmware version (Major)	UINT	Firmware version (Major)
	24	8	Firmware version (Minor)	UINT	Firmware version (Minor)
	32	16	Runtime (seconds)	UINT	Number of seconds since processor start-up
	48	16	Runtime (milliseconds)	UINT	Number of milliseconds (after the integer second) since processor start-up

Table 28: Boot And Running Program Status Telemetry Format

ID	129		Frame Length (bytes)		6
Description	Boot And Running Program Status				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	4	Cause of MCU Reset	ENUM	Cause of MCU reset. Possible values are in Table 29: ResetCause Enumeration Values
	4	4	Boot Cause	ENUM	Cause of last reboot. Possible values are in Table 30: BootCause Enumeration Values
	8	16	Boot Counter	UINT	Number of times CubeComputer has booted
	24	8	Boot Program Index	ENUM	Location of program that is currently running. Possible values are in Table 31: BootProgramsList Enumeration Values
	32	8	Firmware version (Major)	UINT	Firmware version (Major)
	40	8	Firmware version (Minor)	UINT	Firmware version (Minor)

Table 29: ResetCause Enumeration Values

Numeric Value	Name	Description
0	Power-On Reset	Power-On Reset
1	Brown-Out Detected on Regulated Power	Brown-Out Detected on Regulated Power
2	Brown-Out Detected on Unregulated Power	Brown-Out Detected on Unregulated Power
3	External Watchdog Reset	External Watchdog Reset
4	External Reset	External Reset
5	Watchdog Reset	Watchdog Reset
6	Lockup System Reset	Lockup System Reset
7	Lockup Reset	Lockup Reset
8	System Request Reset	System Request Reset
9	Backup Brown-Out	Backup domain brown-out reset
10	Backup Mode Reset	Backup mode reset
11	Backup Mode RST and Backup Brown-Out Vdd Regulated	Backup Mode reset and Backup domain brown-out on VDD regulated
12	BackupModeRST and Backup Brown-Out Vdd Regulated and Brown-Out Regulated	Backup Mode reset and Backup domain brown-out on VDD regulated and brown out on regulated
13	Backup Mode RST and Watchdog Reset	Backup mode reset and Watchdog reset
14	Backup Brown-Out Buvin and System Request Reset	Backup Domain brown-out on BUVIN and System request reset.
15	26nitial Reset Cause	26nitial Reset Cause

Table 30: BootCause Enumeration Values

Numeric Value	Name	Description
0	Unexpected	Unexpected reset
1	Not Used	
2	Communications Timeout	Timeout due to lack of communications
3	Commanded	Software commanded MCU reset
4	Not Used	
5	SRAM Latchup	Latchup detected in SRAM

Table 31: BootProgramsList Enumeration Values

Numeric Value	Name	Description

1	Internal Flash Program	Internal Flash Program
2	Bootloader	Bootloader

Table 32: Boot Index and Status Telemetry Format

ID	130		**Frame Length (bytes)**		2
Description	Current selected boot index and status of last boot				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Program Index	ENUM	Program Index. Possible values are in Table 31: BootProgramsList Enumeration Values
	8	8	Boot Status	ENUM	Boot Status. Possible values are in Table 33: BootStatus Enumeration Values

Table 33: BootStatus Enumeration Values

Numeric Value	Name	Description
0	New Selection	New Selection
1	Boot Success	Boot Success
2	1 Failed boot attempt	1 Failed boot attempt
3	2 Failed boot attempts	2 Failed boot attempts
4	3 Failed boot attempts	3 Failed boot attempts

Table 34: Last Logged Event Telemetry Format

ID	141		**Frame Length (bytes)**		6
Description	Last Logged Event (relative to pointer – adjusted via Advance and Reset TCs (3 & 4)				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	Time of Event	UINT	
	32	8	Event ID	UINT	Event ID
	40	8	Event Parameter	UINT	Event Parameter

Table 35: SRAM Latchup counters Telemetry Format

ID	142	**Frame Length (bytes)**	6
Description	SRAM Latchup counters		

Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	SRAM1 latchups	UINT	The number of SRAM1 latchups detected
	16	16	SRAM2 latchups	UINT	The number of SRAM2 latchups detected

Table 36: EDAC Error Counters Telemetry Format

ID	143	Frame Length (bytes)		6
Description	EDAC Error Counters			
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Single SRAM upsets	UINT	The number of single SRAM upsets (per byte) detected
	16	16	Double SRAM upsets	UINT	The number of double SRAM upsets (per byte) detected
	32	16	Multiple SRAM upsets	UINT	The number of multiple SRAM upsets (per byte) detected

Table 37: Communication Status Telemetry Format

ID	144	Frame Length (bytes)		6
Description	Communication status – includes command and telemetry counters and error flags			
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Telecommand counter	UINT	No. of telecommands received
	16	16	Telemetry request counter	UINT	No. of telemetry requests received
	32	1	Telecommand buffer overrun	BOOL	TC buffer was overrun while receiving a telecommand
	33	1	UART protocol error	BOOL	UART protocol error occurred
	34	1	UART incomplete message	BOOL	UART start-of-message identifier was received without a preceding end-of-message
	35	1	I2C telemetry error	BOOL	Number of data clocked out was more than telemetry package
	36	1	I2C telecommand buffer error	BOOL	Telecommand sent exceeds buffer size

	37	1	CAN telecommand buffer error	BOOL	Telecommand sent exceeds buffer size

Table 38: SD card format/erase progress Telemetry Format

ID	234		**Frame Length (bytes)**		1
Description	SD card format or erase progress				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	1	Format Busy	BOOL	Busy formatting SD card
	1	1	Erase All Busy	BOOL	Busy formatting erasing all SD files

Table 39: Telecommand Acknowledge Telemetry Format

ID	240		**Frame Length (bytes)**		4
Description	Telemetry frame with acknowledge status of the previously sent command				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Last TC ID	UINT	ID of last received TC
	8	1	Processed flag	BOOL	Flag to indicate if the last TC has been processed.
	16	8	TC error status	ENUM	Status of last processed telecommand. Possible values are in Table 40: TcErrorReason Enumeration Values
	24	8	TC parameter error index	UINT	Index of incorrect TC parameter

Table 40: TcErrorReason Enumeration Values

Numeric Value	Name	Description
0	No Error	No error
1	Invalid TC	Invalid telecommand ID
2	Incorrect Length	Incorrect TC parameter length
3	Incorrect Parameter	Incorrect TC parameter value
4	CRC check failed	CRC check failed

Table 41: File Download Buffer with File Contents Telemetry Format

ID	241	**Frame Length (bytes)**	22

Description	File Download buffer 20-byte packet				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Packet counter	UINT	Packet counter of this file download packet
	16	160	File bytes	ARRAY	File 20-byte packet

Table 42: Download Block Ready Telemetry Format

ID	242		Frame Length (bytes)		5
Description	Status about download block preparation				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	1	Ready	BOOL	Ready
	1	1	ParameterError	BOOL	The combination of message length and hole map resulted in invalid array lengths
	8	16	Block CRC16 Checksum	UINT	Block CRC16 Checksum
	24	16	Block Length	UINT	Block length

Table 43: File Information Telemetry Format

ID	243		Frame Length (bytes)		12
Description	File Information				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	4	File Type	ENUM	File Type. Possible values are in Table 16: FileType Enumeration Values
	4	1	Busy Updating	BOOL	
	8	8	File Ctr	UINT	File Counter
	16	32	File Size	UINT	File Size
	48	32	File Date and Time	UINT	File Date and Time (in MS-DOS format). (Unit of measure is [s])
	80	16	File CRC16 Checksum	UINT	File CRC16 Checksum

Table 44: Initialize Upload Complete Telemetry Format

ID	244		Frame Length (bytes)		1
Description	Initialize Upload Complete				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	1	Busy	BOOL	Busy with file initialization (flash erase, etc.)

Table 45: Upload Block Complete Telemetry Format

ID	245		Frame Length (bytes)		1
Description	Finalize Upload Block Complete				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	1	Busy	BOOL	Busy with block finalization
	1	1	Error	BOOL	Error in block finalization

Table 46: Block Checksum Telemetry Format

ID	246		Frame Length (bytes)		2
Description	File upload Block CRC16 Checksum				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Checksum	UINT	Checksum

5.1.3 Common Configuration Messages

Table 47: List of Common Configuration Messages

Set ID	Get ID	Name	Description	Length (bytes)
General				
3	131	Cache enabled state	Cache enabled state – Table 48: Cache enabled state Message Format	1
8	134	SRAM Scrub Parameters	SRAM scrubbing size – Table 49: SRAM Scrub Parameters Message Format	2
9	145	Unix Time Save to Flash	Configuration settings for Unix time flash memory persistence – Table 51: Unix Time Save to Flash Message Format	2
120	247	Hole Map 1	File Upload Hole Map 1 – Table 52: Hole Map 1 Message Format	16

121	248	Hole Map 2	File Upload Hole Map 2 – Table 53: Hole Map 2 Message Format	16
122	249	Hole Map 3	File Upload Hole Map 3 – Table 54: Hole Map 3 Message Format	16
123	250	Hole Map 4	File Upload Hole Map 4 – Table 55: Hole Map 4 Message Format	16
124	251	Hole Map 5	File Upload Hole Map 5 – Table 56: Hole Map 5 Message Format	16
125	252	Hole Map 6	File Upload Hole Map 6 – Table 57: Hole Map 6 Message Format	16
126	253	Hole Map 7	File Upload Hole Map 7 – Table 58: Hole Map 7 Message Format	16
127	254	Hole Map 8	File Upload Hole Map 8 – Table 59: Hole Map 8 Message Format	16
Satellite State				
2	140	Current Unix Time	Current Unix Time – Table 50: Current Unix Time Message Format	6

Table 48: Cache enabled state Message Format

Set ID/Get ID	3/131		**Parameters Length (bytes)**		1
Description	Cache enabled state				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	1	Enabled state	BOOL	Enabled state

Table 49: SRAM Scrub Parameters Message Format

Set ID/Get ID	8/134		**Parameters Length (bytes)**		2
Description	SRAM scrubbing size				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Scrub Size	UINT	Scrub Size

Table 50: Current Unix Time Message Format

Set ID/Get ID	2/140		**Parameters Length (bytes)**		6
Description	Current Unix Time				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	Current Unix Time	UINT	Time in s since 01/01/1970, 00:00. (Unit of measure is [s])

	32	16	Milliseconds	UINT	Current millisecond count. (Unit of measure is [ms])

Table 51: Unix Time Save to Flash Message Format

Set ID/Get ID	9/145		**Parameters Length (bytes)**		2
Description	Configuration settings for unixtime flash memory persistence				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	1	Save Now	BOOL	Save current Unix time to flash memory
	1	1	Save On Update	BOOL	Save Unix time to flash memory whenever there is a command to update the Unix time
	2	1	Save Periodic	BOOL	Save Unix time to flash memory periodically
	8	8	Period	UINT	Interval at which to save Unix time to flash memory. (Unit of measure is [s])

Table 52: Hole Map 1 Message Format

Set ID/Get ID	120/247		**Parameters Length (bytes)**		16
Description	File Upload Hole Map 1				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	128	Hole Map	ARRAY	Hole Map

Table 53: Hole Map 2 Message Format

Set ID/Get ID	121/248		**Parameters Length (bytes)**		16
Description	File Upload Hole Map 2				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	128	Hole Map	ARRAY	Hole Map

Table 54: Hole Map 3 Message Format

Set ID/Get ID	122/249		**Parameters Length (bytes)**		16
Description	File Upload Hole Map 3				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	128	Hole Map	ARRAY	Hole Map

Table 55: Hole Map 4 Message Format

Set ID/Get ID	123/250		Parameters Length (bytes)		16
Description	File Upload Hole Map 4				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	128	Hole Map	ARRAY	Hole Map

Table 56: Hole Map 5 Message Format

Set ID/Get ID	124/251		Parameters Length (bytes)		16
Description	File Upload Hole Map 5				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	128	Hole Map	ARRAY	Hole Map

Table 57: Hole Map 6 Message Format

Set ID/Get ID	125/252		Parameters Length (bytes)		16
Description	File Upload Hole Map 6				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	128	Hole Map	ARRAY	Hole Map

Table 58: Hole Map 7 Message Format

Set ID/Get ID	126/253		Parameters Length (bytes)		16
Description	File Upload Hole Map 7				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	128	Hole Map	ARRAY	Hole Map

Table 59: Hole Map 8 Message Format

Set ID/Get ID	127/254		Parameters Length (bytes)		16
Description	File Upload Hole Map 8				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	128	Hole Map	ARRAY	Hole Map

5.2 BootLoader

5.2.1 BootLoader Telecommands

Table 60: List of BootLoader Telecommands

ID	Name	Description	Length (bytes)
General			
7	Clear Error Flags	Clear Error Flags – Table 61: Clear Error Flags Command Format	0
100	Set Boot Index	Select which program to boot – Table 62: Set Boot Index Command Format	1
101	Run Selected Program	Run Selected Program – Table 64: Run Selected Program Command Format	0
102	Read Program Information	Read Program CRC, length – Table 65: Read Program Information Command Format	1
103	Copy Program to Internal Flash	Copy Program to Internal Flash – Table 67: Copy Program to Internal Flash Command Format	2

Table 61: Clear Error Flags Command Format

ID	7	**Parameters Length (bytes)**	0
Description	Clear Error Flags		

Table 62: Set Boot Index Command Format

ID	100		**Parameters Length (bytes)**		1
Description	Select which program to boot				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Program Index	ENUM	Program Index. Possible values are in Table 63: BootSetProgramsList Enumeration Values

Table 63: BootSetProgramsList Enumeration Values

Numeric Value	Name	Description
1	Internal Flash Program	Internal Flash Program

Table 64: Run Selected Program Command Format

ID	101	**Parameters Length (bytes)**	0
Description	Run Selected Program		

Table 65: Read Program Information Command Format

ID	102		**Parameters Length (bytes)**		1
Description	Read Program CRC, length				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Program Index	ENUM	Program Index. Possible values are in Table 66: ProgramsList Enumeration Values

Table 66: ProgramsList Enumeration Values

Numeric Value	Name	Description
0	Bootloader	Bootloader
1	Internal Flash Program	Internal Flash Program
2	EEPROM	EEPROM
3	External Flash Program 1	External Flash Program 1
4	External Flash Program 2	External Flash Program 2
5	External Flash Program 3	External Flash Program 3
6	External Flash Program 4	External Flash Program 4
7	External Flash Program 5	External Flash Program 5
8	External Flash Program 6	External Flash Program 6
9	External Flash Program 7	External Flash Program 7
10	SD User file 1	SD User file 1
11	SD User file 2	SD User file 2
12	SD User file 3	SD User file 3
13	SD User file 4	SD User file 4
14	SD User file 5	SD User file 5
15	SD User file 6	SD User file 6
16	SD User file 7	SD User file 7
17	SD User file 8	SD User file 8

Table 67: Copy Program to Internal Flash Command Format

ID	103	**Parameters Length (bytes)**	2

Description	Copy Program to Internal Flash				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	Source Program Index	ENUM	Source Program Index. Possible values are in Table 66: ProgramsList Enumeration Values
	8	8	Bootloader overwrite flag	UINT	Bootloader overwrite flag. Set to 0x5A to overwrite the boot segment. USE WITH CAUTION!

5.2.2 BootLoader Telemetry

Table 68: List of BootLoader Telemetry Frames

ID	Name	Description	Length (bytes)
General			
132	Bootloader State	Status flags for bootloader – Table 69: Bootloader State Telemetry Format	6
232	Program Information	Program information including file size and CRC – Table 70: Program Information Telemetry Format	8
233	Copy To Internal Flash Progress	Progress of copy to internal flash operation – Table 71: Copy To Internal Flash Progress Telemetry Format	1

Table 69: Bootloader State Telemetry Format

ID	132	Frame Length (bytes)		6
Description	Status flags for bootloader			
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Up-time (s)	UINT	Uptime. (Unit of measure is [s])
	16	1	SRAM1 is enabled	BOOL	SRAM1 is enabled
	17	1	SRAM2 is enabled	BOOL	SRAM2 is enabled
	18	1	SRAM Latch-up Error occurred and could not be recovered	BOOL	SRAM Latch-up Error occurred and could not be recovered
	19	1	SRAM Latch-up Occurred but	BOOL	SRAM Latch-up Occurred but recovered after power cycle

			recovered after power cycle		
	20	1	SD card 38nitialization error	BOOL	Flag to indicate that the SD card failed to initialise
	21	1	SD card read error	BOOL	Flag to indicate that a read operation from the SD card failed
	22	1	SD card write error	BOOL	Flag to indicate that a write operation to the SD card failed
	23	1	External Flash Error	BOOL	External Flash Erase/write error occurred
	24	1	Internal Flash Error	BOOL	Internal Flash Erase/write error occurred
	25	1	EEPROM Error	BOOL	EEPROM Write error occurred
	26	1	Boot Register Corrupt	BOOL	Boot Register contained invalid data
	27	1	Communications Error with Radio	BOOL	Communications Error with Radio

Table 70: Program Information Telemetry Format

ID	232		**Frame Length (bytes)**		8
Description	Program information including file size and CRC				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Program Index	ENUM	Program Index. Possible values are in Table 66: ProgramsList Enumeration Values
	8	1	Busy	BOOL	Busy reading
	16	32	File Size	UINT	File Size (bytes)
	48	16	Checksum	UINT	CRC16 Checksum

Table 71: Copy To Internal Flash Progress Telemetry Format

ID	233		**Frame Length (bytes)**		1
Description	Progress of copy to internal flash operation				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	1	Busy	BOOL	Busy
	1	1	Error	BOOL	Error

5.3 CubeAcp

5.3.1 CubeAcp Telecommands

Table 72: List of CubeAcp Telecommands

ID	Name	Description	Length (bytes)
General			
7	Deploy Magnetometer Boom	Deploy magnetometer boom – Table 73: Deploy Magnetometer Boom Command Format	1
10	ADCS Run Mode	Set ADCS enabled state & control loop behaviour – Table 74: ADCS Run Mode Command Format	1
12	Clear Errors	Clear Latched Error Flags – Table 76: Clear Errors Command Format	1
13	Set Attitude Control Mode	Set attitude control mode – Table 77: Set Attitude Control Mode Command Format	3
14	Set Attitude Estimation Mode	Set attitude estimation mode – Table 79: Set Attitude Estimation Mode Command Format	1
18	Trigger ADCS Loop	Trigger ADCS to perform one iteration of the control loop (only valid when ADCS Run Mode is Triggered) – Table 83: Trigger ADCS Loop Command Format	0
19	Trigger ADCS Loop with Simulated Sensor Data	Trigger ADCS to perform one iteration of the control loop (only valid when ADCS Run Mode is Triggered) – Table 84: Trigger ADCS Loop with Simulated Sensor Data Command Format	127
31	ASGP4 Run Mode	Set ASGP4 run mode – Table 86: ASGP4 Run Mode Command Format	1
32	ASGP4 Trigger	Trigger a start of the ASGP4 process – Table 88: ASGP4 Trigger Command Format	0
56	Set Mode of Magnetometer Operation	Use of main or redundant magnetometer – Table 89: Set Mode of Magnetometer Operation Command Format	1
57	Convert to JPG file	Convert raw or bmp files to JPG – Table 91: Convert to JPG file Command Format	3
80	Save Image	Save and capture image from one of CubeSense cameras or CubeStar camera to SD card – Table 94: Save Image Command Format	2
Actuator Commands			
16	Set Magnetorquer Output	Set magnetorquer output (only valid if Control Mode is None) – Table 81: Set Magnetorquer Output Command Format	6
17	Set Wheel Speed	Set wheel speed (only valid if Control Mode is None) – Table 82: Set Wheel Speed Command Format	6

Configuration			
63	Save Configuration	Save current configuration to flash memory – Table 92: Save Configuration Command Format	0
64	Save Orbit Parameters	Save current orbit parameters to flash memory – Table 93: Save Orbit Parameters Command Format	0

Table 73: Deploy Magnetometer Boom Command Format

ID	7		**Parameters Length (bytes)**		1
Description	Deploy magnetometer boom				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Timeout	UINT	Deployment actuation timeout value. (Unit of measure is [s])

Table 74: ADCS Run Mode Command Format

ID	10		**Parameters Length (bytes)**		1
Description	Set ADCS enabled state & control loop behaviour				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Enabled	ENUM	Set ADCS enabled state. When disabled the CubeACP will not use the ADCS I2C bus. Possible values are in Table 75: AdcsRunMode Enumeration Values

Table 75: AdcsRunMode Enumeration Values

Numeric Value	Name	Description
0	Off	ADCS loop is inactive
1	Enabled	ADCS 1Hz loop is active
2	Triggered	ADCS will execute control loop only when triggered
3	Simulation	ADCS is in simulation mode

Table 76: Clear Errors Command Format

ID	12	**Parameters Length (bytes)**	1
Description	Clear Latched Error Flags		

Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	1	ADCS Error Flags	BOOL	Clear ADCS error flags
	1	1	HK Error Flags	BOOL	Clear HK Error flags

Table 77: Set Attitude Control Mode Command Format

ID	13		Parameters Length (bytes)	3
Description	Set attitude control mode			
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	Control Mode	ENUM	Attitude control mode. Possible values are in Table 78: ConModeSelect Enumeration Values
	8	16	Timeout	UINT	Control timeout duration. Control will revert to None when timer reaches zero. 0xFFFF for infinite timeout. (Unit of measure is [s])

Table 78: ConModeSelect Enumeration Values

Numeric Value	Name	Description
0	No control	No control
1	Detumbling control	Detumbling control
2	Y-Thomson spin	Y-Thomson spin
3	Y-Wheel momentum stabilized – Initial Pitch Acquisition	Y-Wheel momentum stabilized – Initial Pitch Acquisition
4	Y-Wheel momentum stabilized – Steady State	Y-Wheel momentum stabilized – Steady State
5	XYZ-Wheel control	XYZ-Wheel control
6	Rwheel sun tracking control	Rwheel sun tracking control
7	Rwheel target tracking control	Rwheel target tracking control
8	Very Fast-spin Detumbling control	10Hz Detumbling control within CubeControl
9	Fast-spin Detumbling control	Fast Detumbling control
10	User Specific Control Mode 1	User defined, or custom control mode 1
11	User Specific Control Mode 2	User defined, or custom control mode 2
12	Stop R-wheels	Stop all R-wheels
13	User Coded Control Mode	User coded control mode
14	Sun-tracking yaw- or roll-only wheel control mode	Yaw-only or roll-only sun tracking mode.

15	Target-tracking yaw-only wheel control Mode	Yaw-only wheel control to align selected facet to ground target

Table 79: Set Attitude Estimation Mode Command Format

ID	14		**Parameters Length (bytes)**		1
Description	Set attitude estimation mode				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Attitude Estimation Mode	ENUM	Attitude estimation mode. Possible values are in Table 80: EstimModeSelect Enumeration Values

Table 80: EstimModeSelect Enumeration Values

Numeric Value	Name	Description
0	No attitude estimation	No attitude estimation
1	MEMS rate sensing	MEMS rate sensing
2	Magnetometer rate filter	Magnetometer rate filter
3	Magnetometer rate filter with pitch estimation	Magnetometer rate filter with pitch estimation
4	Magnetometer and Fine-sun TRIAD algorithm	Magnetometer and Fine-sun TRIAD algorithm
5	Full-state EKF	Full-state EKF
6	MEMS gyro EKF	MEMS gyro EKF
7	User Coded Estimation Mode	User coded estimation mode

Table 81: Set Magnetorquer Output Command Format

ID	16		**Parameters Length (bytes)**		6
Description	Set magnetorquer output (only valid if Control Mode is None)				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Commanded X Magnetorquer duty cycle	INT	Commanded X-torquer duty cycle. Raw parameter value is obtained using the formula: (raw parameter) = (formatted value)*1000.0

	16	16	Commanded Y Magnetorquer duty cycle	INT	Commanded Y-torquer duty cycle. Raw parameter value is obtained using the formula: (raw parameter) = (formatted value)*1000.0
	32	16	Commanded Z Magnetorquer duty cycle	INT	Commanded Z-torquer duty cycle. Raw parameter value is obtained using the formula: (raw parameter) = (formatted value)*1000.0

Table 82: Set Wheel Speed Command Format

ID	17		**Parameters Length (bytes)**		6
Description	Set wheel speed (only valid if Control Mode is None)				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Commanded X speed	INT	Commanded X-wheel speed. (Unit of measure is [rpm])
	16	16	Commanded Y speed	INT	Commanded Y-wheel speed. (Unit of measure is [rpm])
	32	16	Commanded Z speed	INT	Commanded Z-wheel speed. (Unit of measure is [rpm])

Table 83: Trigger ADCS Loop Command Format

ID	18	**Parameters Length (bytes)**	0
Description	Trigger ADCS to perform one iteration of the control loop (only valid when ADCS Run Mode is Triggered)		

Table 84: Trigger ADCS Loop with Simulated Sensor Data Command Format

ID	19		**Parameters Length (bytes)**		127
Description	Trigger ADCS to perform one iteration of the control loop (only valid when ADCS Run Mode is Triggered)				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	Unix Time	UINT	Unix time for iteration. (Unit of measure is [s])
	32	16	Css Raw 1	UINT	CSS1 raw measurement
	48	16	Css Raw 2	UINT	CSS2 raw measurement

	64	16	Css Raw 3	UINT	CSS3 raw measurement
	80	16	Css Raw 4	UINT	CSS4 raw measurement
	96	16	Css Raw 5	UINT	CSS5 raw measurement
	112	16	Css Raw 6	UINT	CSS6 raw measurement
	128	16	Css Raw 7	UINT	CSS7 raw measurement
	144	16	Css Raw 8	UINT	CSS8 raw measurement
	160	16	Css Raw 9	UINT	CSS9 raw measurement
	176	16	Css Raw 10	UINT	CSS10 raw measurement
	192	16	Cam1 Raw X	INT	Cam1 sensor raw X angle
	208	16	Cam1 Raw Y	INT	Cam1 sensor raw Y angle
	224	8	Cam1 Busy	UINT	Cam1 sensor capture status
	232	8	Cam1 Result	UINT	Cam1 sensor detection result
	240	16	Cam2 Raw X	INT	Cam2 sensor raw X angle
	256	16	Cam2 Raw Y	INT	Cam2 sensor raw Y angle
	272	8	Cam2 Busy	UINT	Cam2 sensor capture status
	280	8	Cam2 Result	UINT	Cam2 sensor detection result
	288	16	Mag Raw X	INT	Raw magnetometer X measurement
	304	16	Mag Raw Y	INT	Raw magnetometer Y measurement
	320	16	Mag Raw Z	INT	Raw magnetometer Z measurement
	336	32	Rate Raw X	INT	Raw X rate sensor measurement
	368	32	Rate Raw Y	INT	Raw Y rate sensor measurement
	400	32	Rate Raw Z	INT	Raw Z rate sensor measurement
	432	16	Wheel Raw X	INT	Raw X wheel speed measurement. (Unit of measure is [rpm])
	448	16	Wheel Raw Y	INT	Raw Y wheel speed measurement. (Unit of measure is [rpm])
	464	16	Wheel Raw Z	INT	Raw Z wheel speed measurement. (Unit of measure is [rpm])
	480	16	Star1CameraX	INT	Star1 camera X-vector
	496	16	Star1CameraY	INT	Star1 camera Y-vector
	512	16	Star1CameraZ	INT	Star1 camera Z-vector
	528	16	Star1InertialX	INT	Star1 inertial X-vector
	544	16	Star1InertialY	INT	Star1 inertial Y-vector
	560	16	Star1InertialZ	INT	Star1 inertial Z-vector
	576	16	Star2CameraX	INT	Star2 camera X-vector

	592	16	Star2CameraY	INT	Star2 camera Y-vector
	608	16	Star2CameraZ	INT	Star2 camera Z-vector
	624	16	Star2InertialX	INT	Star2 inertial X-vector
	640	16	Star2InertialY	INT	Star2 inertial Y-vector
	656	16	Star2InertialZ	INT	Star2 inertial Z-vector
	672	16	Star3CameraX	INT	Star3 camera X-vector
	688	16	Star3CameraY	INT	Star3 camera Y-vector
	704	16	Star3CameraZ	INT	Star3 camera Z-vector
	720	16	Star3InertialX	INT	Star3 inertial X-vector
	736	16	Star3InertialY	INT	Star3 inertial Y-vector
	752	16	Star3InertialZ	INT	Star3 inertial Z-vector
	768	8	Gps Solution Status	ENUM	GPS Solution Status. Possible values are in Table 85: GpsSolutionStatus Enumeration Values
	776	16	GPS Reference Week	UINT	GPS Reference Week
	792	32	GPS Time Milliseconds	UINT	GPS Time Milliseconds. (Unit of measure is [ms])
	824	32	ECEF Position X	INT	ECEF Position X. (Unit of measure is [m])
	856	16	ECEF Velocity X	INT	ECEF Velocity X. (Unit of measure is [m/s])
	872	32	ECEF Position Y	INT	ECEF Position Y. (Unit of measure is [m])
	904	16	ECEF Velocity Y	INT	ECEF Velocity Y. (Unit of measure is [m/s])
	920	32	ECEF Position Z	INT	ECEF Position Z. (Unit of measure is [m])
	952	16	ECEF Velocity Z	INT	ECEF Velocity Z. (Unit of measure is [m/s])
	968	8	X-pos Standard Deviation	UINT	X-pos Standard Deviation. Raw parameter value is obtained using the formula: (raw parameter) = (formatted value)*10.0 (formatted value is in [m] units)
	976	8	Y-pos Standard Deviation	UINT	Y-pos Standard Deviation. Raw parameter value is obtained using the formula: (raw parameter) = (formatted

					value)*10.0 (formatted value is in [m] units)
	984	8	Z-pos Standard Deviation	UINT	Z-pos Standard Deviation. Raw parameter value is obtained using the formula: (raw parameter) = (formatted value)*10.0 (formatted value is in [m] units)
	992	8	X-vel Standard Deviation	UINT	X-vel Standard Deviation. (Unit of measure is [m/s])
	1000	8	Y-vel Standard Deviation	UINT	Y-vel Standard Deviation. (Unit of measure is [m/s])
	1008	8	Z-vel Standard Deviation	UINT	Z-vel Standard Deviation. (Unit of measure is [m/s])

Table 85: GpsSolutionStatus Enumeration Values

Numeric Value	Name	Description
0	Solution computed	Solution computed
1	Insufficient observations	Insufficient observations
2	No convergence	No convergence
3	Singularity at parameters matrix	Singularity at parameters matrix
4	Covariance trace exceeds maximum	Covariance trace exceeds maximum
5	Not yet converged from cold start	Not yet converged from cold start
6	Height or velocity limits exceeded	Height or velocity limits exceeded
7	Variance exceeds limits	Variance exceeds limits
8	Large residuals make position unreliable	Large residuals make position unreliable
9	Calculating comparison to user provided	Calculating comparison to user provided
10	The fixed position is invalid	The fixed position is invalid
11	Position type is unauthorized	Position type is unauthorized

Table 86: ASGP4 Run Mode Command Format

ID	31		**Parameters Length (bytes)**	1
Description	Set ASGP4 run mode			
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**

	0	8	ASGP4 Mode	ENUM	ASGP4 enabled state. Possible values are in Table 87: Asgp4ModeSelect Enumeration Values

Table 87: Asgp4ModeSelect Enumeration Values

Numeric Value	Name	Description
0	Off	ASGP4 is inactive
1	Trigger	ASGP4 is active but waiting for trigger TC
2	Background	ASGP4 runs asynchronously with GPS data received, TLEs updated internal to asgp4 module but not used
3	Augment	ASGP4 runs in background but when processing of data is complete, TLEs used by orbit model is updated with asgp4 estimated TLEs

Table 88: ASGP4 Trigger Command Format

ID	32	**Parameters Length (bytes)**	0
Description	Trigger a start of the ASGP4 process		

Table 89: Set Mode of Magnetometer Operation Command Format

ID	56		**Parameters Length (bytes)**		1
Description	Use of main or redundant magnetometer				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Magnetometer Mode	ENUM	Mode describing which magnetometer is used for estimation and control. Possible values are in Table 90: MagModeVal Enumeration Values

Table 90: MagModeVal Enumeration Values

Numeric Value	Name	Description
0	Main MTM Sampled Through Signal	Main magnetometer, sampled through signal microcontroller

1	Redundant MTM Sampled Through Signal	Redundant magnetometer, sampled through signal microcontroller
2	Main MTM Sampled Through Motor	Main magnetometer, sampled through motor microcontroller
3	None	None

Table 91: Convert to JPG file Command Format

ID	57		**Parameters Length (bytes)**		3
Description	Convert raw or bmp files to JPG				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Source File Counter	UINT	Source File Counter
	8	8	Quality Factor	UINT	Quality Factor
	16	8	White Balance	UINT	White Balance

Table 92: Save Configuration Command Format

ID	63	**Parameters Length (bytes)**	0
Description	Save current configuration to flash memory		

Table 93: Save Orbit Parameters Command Format

ID	64	**Parameters Length (bytes)**	0
Description	Save current orbit parameters to flash memory		

Table 94: Save Image Command Format

ID	80		**Parameters Length (bytes)**		2
Description	Save and capture image from one of CubeSense cameras or CubeStar camera to SD card				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Camera Select	ENUM	Camera Selection. Possible values are in Table 95: CamSelect Enumeration Values
	8	8	Image Size	ENUM	Image size selection. Possible values are in Table 96: ImSize Enumeration Values

Table 95: CamSelect Enumeration Values

Numeric Value	Name	Description
0	Cam1	Cam1 camera
1	Cam2	Cam2 camera
2	Star	Star camera

Table 96: ImSize Enumeration Values

Numeric Value	Name	Description
0	Size 0	1024 x 1024 pixels
1	Size 1	512 x 512 pixels
2	Size 2	256 x 256 pixels
3	Size 3	128 x 128 pixels
4	Size 4	64 x 64 pixels

5.3.2 CubeAcp Telemetry

Table 97: List of CubeAcp Telemetry Frames

ID	Name	Description	Length (bytes)
ADCS State			
132	Current ADCS State	Current state of the Attitude Control Processor – frame 1 – Table 98: Current ADCS State Telemetry Format	6
146	Estimated Attitude Angles	Estimated attitude angles – Table 102: Estimated Attitude Angles Telemetry Format	6
147	Estimated Angular Rates	Estimated angular rates relative to orbit reference frame – Table 103: Estimated Angular Rates Telemetry Format	6
148	Satellite Position (ECI)	Satellite position in ECI frame – Table 104: Satellite Position (ECI) Telemetry Format	6
149	Satellite Velocity (ECI)	Satellite velocity in ECI frame – Table 105: Satellite Velocity (ECI) Telemetry Format	6
150	Satellite Position (LLH)	Satellite position in WGS-84 coordinate frame – Table 106: Satellite Position (LLH) Telemetry Format	6
190	ADCS State	Current ADCS state – Table 149: ADCS State Telemetry Format	54
201	Fine Estimated Angular Rates	High resolution estimated angular rates relative to orbit reference frame – Table 157: Fine Estimated Angular Rates Telemetry Format	6

218	Estimated Quaternion	Estimated quaternion set – Table 165: Estimated Quaternion Telemetry Format	6
219	ECEF Position	Satellite position in ECEF coordinates – Table 166: ECEF Position Telemetry Format	6
224	Current ADCS State 2	Current state of the Attitude Control Processor – frame 2 – Table 169: Current ADCS State 2 Telemetry Format	6
General			
133	JPG Conversion Progress	Conversion progress – Table 99: JPG Conversion Progress Telemetry Format	3
135	CubeACP State	Contains flags regarding the state of the ACP – Table 101: CubeACP State Telemetry Format	1
196	Adcs Execution Times	Returns information about execution times of ACP functions – Table 155: Adcs Execution Times Telemetry Format	8
220	ACP Execution State	Returns information about the ACP loop – Table 167: ACP Execution State Telemetry Format	3
233	Status of Image Capture and Save Operation	Status of Image Capture and Save Operation – Table 176: Status of Image Capture and Save Operation Telemetry Format	2
ADCS Measurements			
151	Magnetic Field Vector	Measured magnetic field vector – Table 107: Magnetic Field Vector Telemetry Format	6
152	Coarse Sun Vector	Measured coarse sun vector – Table 108: Coarse Sun Vector Telemetry Format	6
153	Fine Sun Vector	Measured fine sun vector – Table 109: Fine Sun Vector Telemetry Format	6
154	Nadir Vector	Measured nadir vector – Table 110: Nadir Vector Telemetry Format	6
155	Rate Sensor Rates	Rate sensor measurements – Table 111: Rate Sensor Rates Telemetry Format	6
156	Wheel Speed	Wheel speed measurement – Table 112: Wheel Speed Telemetry Format	6
181	Star 1 Body Vector	Star 1 Body Vector – Table 139: Star 1 Body Vector Telemetry Format	6
182	Star 2 Body Vector	Star 2 Body Vector – Table 140: Star 2 Body Vector Telemetry Format	6
183	Star 3 Body Vector	Star 3 Body Vector – Table 141: Star 3 Body Vector Telemetry Format	6
184	Star 1 Orbit Vector	Star 1 Orbit Vector – Table 142: Star 1 Orbit Vector Telemetry Format	6
185	Star 2 Orbit Vector	Star 2 Orbit Vector – Table 143: Star 2 Orbit Vector Telemetry Format	6

186	Star 3 Orbit Vector	Star 3 Orbit Vector – Table 144: Star 3 Orbit Vector Telemetry Format	6
191	ADCS Measurements	Calibrated sensor measurements – Table 150: ADCS Measurements Telemetry Format	72
Actuator Commands			
157	Magnetorquer Command	Magnetorquer commands – Table 113: Magnetorquer Command Telemetry Format	6
158	Wheel Speed Commands	Wheel speed commands – Table 114: Wheel Speed Commands Telemetry Format	6
192	Actuator Commands	Actuator commands – Table 151: Actuator Commands Telemetry Format	12
Estimation			
159	IGRF Modelled Magnetic Field Vector	IGRF modelled magnetic field vector (orbit frame referenced) – Table 115: IGRF Modelled Magnetic Field Vector Telemetry Format	6
160	Modelled Sun Vector	Modelled sun vector (orbit frame referenced) – Table 116: Modelled Sun Vector Telemetry Format	6
161	Estimated Gyro Bias	Estimated rate sensor bias – Table 117: Estimated Gyro Bias Telemetry Format	6
162	Estimation Innovation Vector	Estimation innovation vector – Table 118: Estimation Innovation Vector Telemetry Format	6
163	Quaternion Error Vector	Quaternion error vector – Table 119: Quaternion Error Vector Telemetry Format	6
164	Quaternion Covariance	Quaternion covariance – Table 120: Quaternion Covariance Telemetry Format	6
165	Angular Rate Covariance	Angular rate covariance – Table 121: Angular Rate Covariance Telemetry Format	6
193	Estimation Data	Estimation meta-data – Table 152: Estimation Data Telemetry Format	42
228	ASGP4 TLEs	ASGP4 TLEs generated – Table 170: ASGP4 TLEs Telemetry Format	33
Raw Sensor Measurements			
166	Raw Cam2 Sensor	Cam2 sensor capture and detection result – Table 122: Raw Cam2 Sensor Telemetry Format	6
167	Raw Cam1 Sensor	Cam1 sensor capture and detection result – Table 125: Raw Cam1 Sensor Telemetry Format	6
168	Raw CSS 1 to 6	Raw CSS measurements 1 to 6 – Table 126: Raw CSS 1 to 6 Telemetry Format	6
169	Raw CSS 7 to 10	Raw CSS measurements 7 to 10 – Table 127: Raw CSS 7 to 10 Telemetry Format	6
170	Raw Magnetometer	Raw magnetometer measurements – Table 128: Raw Magnetometer Telemetry Format	6

176	Raw GPS Status	Raw GPS status – Table 134: Raw GPS Status Telemetry Format	6
177	Raw GPS Time	Raw GPS time – Table 135: Raw GPS Time Telemetry Format	6
178	Raw GPS X	Raw GPS X position and velocity (ECI referenced) – Table 136: Raw GPS X Telemetry Format	6
179	Raw GPS Y	Raw GPS Y position and velocity (ECI referenced) – Table 137: Raw GPS Y Telemetry Format	6
180	Raw GPS Z	Raw GPS Z position and velocity (ECI referenced) – Table 138: Raw GPS Z Telemetry Format	6
187	Star Magnitude	Instrument magnitude of identified stars – Table 145: Star Magnitude Telemetry Format	6
188	Star Performance1	Performance parameters of star measurement – Table 146: Star Performance1 Telemetry Format	6
189	Star Timing	Timing information of star measurement – Table 148: Star Timing Telemetry Format	6
194	Raw Sensor Measurements	Raw sensor measurements – Table 153: Raw Sensor Measurements Telemetry Format	34
210	Raw GPS Measurements	Raw GPS measurements – Table 158: Raw GPS Measurements Telemetry Format	36
211	Raw Star Tracker	Raw Star Tracker Measurement – Table 159: Raw Star Tracker Telemetry Format	54
212	Star 1 Raw Data	Catalogue index and detected coordinates for star 1 – Table 160: Star 1 Raw Data Telemetry Format	6
213	Star 2 Raw Data	Catalogue index and detected coordinates for star 2 – Table 161: Star 2 Raw Data Telemetry Format	6
214	Star 3 Raw Data	Catalogue index and detected coordinates for star 3 – Table 162: Star 3 Raw Data Telemetry Format	6
215	Secondary Magnetometer Raw Measurements	Secondary Magnetometer raw measurements – Table 163: Secondary Magnetometer Raw Measurements Telemetry Format	6
216	Raw Rate Sensor	Raw rate sensor measurements – Table 164: Raw Rate Sensor Telemetry Format	6
229	CubeStar Estimated Rates	Angular rates estimated by CubeStar – Table 172: CubeStar Estimated Rates Telemetry Format	6
230	CubeStar Estimated Quaternion	Attitude quaternion estimated by CubeStar – Table 173: CubeStar Estimated Quaternion Telemetry Format	6
231	Star Performance2	Performance parameters of star measurement – Table 174: Star Performance2 Telemetry Format	6
ADCS Power			

171	CubeSense1 Current Measurements	CubeSense1 current measurements – Table 129: CubeSense1 Current Measurements Telemetry Format	4
172	CubeControl Current Measurements	CubeControl current measurements – Table 130: CubeControl Current Measurements Telemetry Format	6
173	Wheel Currents	XYZ Wheel current measurement – Table 131: Wheel Currents Telemetry Format	6
174	ADCS Temperatures	Magnetometer + MCU temperature measurements – Table 132: ADCS Temperatures Telemetry Format	6
175	Rate sensor temperatures	Rate sensor temperatures – Table 133: Rate sensor temperatures Telemetry Format	6
195	Power and Temperature Measurements	Power and temperature measurements – Table 154: Power and Temperature Measurements Telemetry Format	38
198	ADCS Misc Current Measurements	CubeStar and Torquer current and temperature measurements – Table 156: ADCS Misc Current Measurements Telemetry Format	6
232	CubeSense2 Current Measurements	CubeSense2 current measurements – Table 175: CubeSense2 Current Measurements Telemetry Format	4

Table 98: Current ADCS State Telemetry Format

ID	132		**Frame Length (bytes)**		6
Description	Current state of the Attitude Control Processor – frame 1				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	4	Attitude Estimation Mode	ENUM	Current attitude estimation mode. Possible values are in Table 80: EstimModeSelect Enumeration Values
	4	4	Control Mode	ENUM	Current attitude control mode. Possible values are in Table 78: ConModeSelect Enumeration Values
	8	2	ADCS Run Mode	ENUM	Current ADCS Running mode. Possible values are in Table 75: AdcsRunMode Enumeration Values
	10	2	ASGP4 Mode	ENUM	ASGP4 enabled state. Possible values are in Table 87:

					Asgp4ModeSelect Enumeration Values
	12	1	CubeControl Signal Enabled	BOOL	CubeControl Signal electronics enabled status
	13	1	CubeControl Motor Enabled	BOOL	CubeControl Motor electronics enabled status
	14	1	CubeSense1 Enabled	BOOL	CubeSense1 enabled status
	15	1	CubeSense2 Enabled	BOOL	CubeSense2 enabled status
	16	1	CubeWheel1 Enabled	BOOL	CubeWheel1 enabled status
	17	1	CubeWheel2 Enabled	BOOL	CubeWheel2 enabled status
	18	1	CubeWheel3 Enabled	BOOL	CubeWheel3 enabled status
	19	1	CubeStar Enabled	BOOL	CubeStar enabled status
	20	1	GPS Receiver Enabled	BOOL	GPS Receiver enabled status
	21	1	GPS LNA Power Enabled	BOOL	GPS Antenna LNA enabled status
	22	1	Motor Driver Enabled	BOOL	Motor Driver Electronics enabled status
	23	1	Sun is Above Local Horizon	BOOL	Sun is above the local horizon (elevation > 0)
	24	1	CubeSense1 Communications Error	BOOL	Communication error occurred with the CubeSense1
	25	1	CubeSense2 Communications Error	BOOL	Communication error occurred with the CubeSense2
	26	1	CubeControl Signal Communications Error	BOOL	Communication error occurred with the CubeControl Signal MCU
	27	1	CubeControl Motor Communications Error	BOOL	Communication error occurred with the CubeControl Motor MCU
	28	1	CubeWheel1 Communications Error	BOOL	Communication error occurred with the CubeWheel1

	29	1	CubeWheel2 Communications Error	BOOL	Communication error occurred with the CubeWheel2
	30	1	CubeWheel3 Communications Error	BOOL	Communication error occurred with the CubeWheel3
	31	1	CubeStar Communications Error	BOOL	Communication error occurred with the CubeStar
	32	1	Magnetometer Range Error	BOOL	Magnetometer measured magnetic field with size 65 uT
	33	1	Cam1 SRAM Overcurrent Detected	BOOL	Cam1 SRAM overcurrent detected
	34	1	Cam1 3V3 Overcurrent Detected	BOOL	Cam1 3V3 overcurrent detected
	35	1	Cam1 Sensor Busy Error	BOOL	Cam1 sensor was not idle at the start of ADCS loop
	36	1	Cam1 Sensor Detection Error	BOOL	Cam1 sensor was unable to compute angles (could be not in FOV)
	37	1	Sun Sensor Range Error	BOOL	Detected sun angles were outside of +/- 90 deg
	38	1	Cam2 SRAM Overcurrent Detected	BOOL	Cam2 SRAM overcurrent detected
	39	1	Cam2 3V3 Overcurrent Detected	BOOL	Cam2 3V3 overcurrent detected
	40	1	Cam2 Sensor Busy Error	BOOL	Cam2 sensor was not idle at the start of ADCS loop
	41	1	Cam2 Sensor Detection Error	BOOL	Cam2 sensor was unable to compute angles (could be not in FOV)
	42	1	Nadir Sensor Range Error	BOOL	Detected nadir angles were outside of +/- 60 deg
	43	1	Rate Sensor Range Error	BOOL	Measured XYZ-body rate is outside of the range +/-20 deg/s
	44	1	Wheel Speed Range Error	BOOL	Wheel XYZ speed measurement was outside the range +/-8500 rpm

	45	1	Coarse Sun Sensor Error	BOOL	Unable to compute Coarse Sun vector (could be not in FOV)
	46	1	StarTracker Match Error	BOOL	Unable to obtain enough matched stars
	47	1	Star Tracker Overcurrent Detected	BOOL	Star tracker overcurrent detected

Table 99: JPG Conversion Progress Telemetry Format

ID	133		**Frame Length (bytes)**		3
Description	Conversion progress				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Progress Percentage	UINT	Progress %
	8	8	Conversion Result	ENUM	JPG Conversion Result. Possible values are in Table 100: JpgConvertResult Enumeration Values
	16	8	Output File Counter	UINT	Output File Counter

Table 100: JpgConvertResult Enumeration Values

Numeric Value	Name	Description
0	Nothing Converted Yet	Nothing Converted Yet
1	Success	Success
2	File Load Error	File Load Error
3	Busy	Busy

Table 101: CubeACP State Telemetry Format

ID	135		**Frame Length (bytes)**		1
Description	Contains flags regarding the state of the ACP				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	1	ADCS Config Load Error	BOOL	Flag to indicate that the ADCS configuration was not read successfully out of flash

	1	1	Orbit Parameter Load Error	BOOL	Flag to indicate that the Orbit parameters were not read successfully out of flash
	2	1	System Configuration Load Error	BOOL	Flag to indicate that the System Configuration was not read successfully out of flash
	3	1	SD card 57nitialization error	BOOL	Flag to indicate that the SD card failed to initialise
	4	1	SD card read error	BOOL	Flag to indicate that a read operation from the SD card failed
	5	1	SD card write error	BOOL	Flag to indicate that a write operation to the SD card failed

Table 102: Estimated Attitude Angles Telemetry Format

ID	146		**Frame Length (bytes)**		6
Description	Estimated attitude angles				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Estimated Roll Angle	INT	Estimated roll angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	16	16	Estimated Pitch Angle	INT	Estimated pitch angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	32	16	Estimated Yaw Angle	INT	Estimated yaw angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01

Table 103: Estimated Angular Rates Telemetry Format

ID	147		**Frame Length (bytes)**		6
Description	Estimated angular rates relative to orbit reference frame				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Estimated X Angular Rate	INT	Estimated X angular rate. Formatted value is obtained

					using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	16	16	Estimated Y Angular Rate	INT	Estimated Y angular rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	32	16	Estimated Z Angular Rate	INT	Estimated Z angular rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01

Table 104: Satellite Position (ECI) Telemetry Format

ID	148		**Frame Length (bytes)**		6
Description	Satellite position in ECI frame				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	X position	INT	ECI referenced X coordinate. Formatted value is obtained using the formula: (formatted value) [km] = RAWVAL*0.25
	16	16	Y position	INT	ECI referenced Y coordinate. Formatted value is obtained using the formula: (formatted value) [km] = RAWVAL*0.25
	32	16	Z position	INT	ECI referenced Z coordinate. Formatted value is obtained using the formula: (formatted value) [km] = RAWVAL*0.25

Table 105: Satellite Velocity (ECI) Telemetry Format

ID	149		**Frame Length (bytes)**		6
Description	Satellite velocity in ECI frame				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	X Velocity	INT	ECI referenced X velocity. Formatted value is obtained using the formula: (formatted value) [m/s] = RAWVAL*0.25
	16	16	Y Velocity	INT	ECI referenced Y velocity. Formatted value is obtained

					using the formula: (formatted value) [m/s] = RAWVAL*0.25
	32	16	Z Velocity	INT	ECI referenced Z velocity. Formatted value is obtained using the formula: (formatted value) [m/s] = RAWVAL*0.25

Table 106: Satellite Position (LLH) Telemetry Format

ID	150		**Frame Length (bytes)**		6
Description	Satellite position in WGS-84 coordinate frame				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Latitude	INT	WGS-84 Latitude angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	16	16	Longitude	INT	Longitude angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	32	16	Altitude	UINT	WGS-84 altitude. Formatted value is obtained using the formula: (formatted value) [km] = RAWVAL*0.01

Table 107: Magnetic Field Vector Telemetry Format

ID	151		**Frame Length (bytes)**		6
Description	Measured magnetic field vector				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Magnetic Field X	INT	Magnetic Field X. Formatted value is obtained using the formula: (formatted value) [uT] = RAWVAL*0.01
	16	16	Magnetic Field Y	INT	Magnetic Field Y. Formatted value is obtained using the formula: (formatted value) [uT] = RAWVAL*0.01
	32	16	Magnetic Field Z	INT	Magnetic Field Z. Formatted value is obtained using the formula:

					(formatted value) [uT] = RAWVAL*0.01

Table 108: Coarse Sun Vector Telemetry Format

ID	152		Frame Length (bytes)		6
Description	Measured coarse sun vector				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Coarse Sun X	INT	Coarse Sun X. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Coarse Sun Y	INT	Coarse Sun Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Coarse Sun Z	INT	Coarse Sun Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 109: Fine Sun Vector Telemetry Format

ID	153		Frame Length (bytes)		6
Description	Measured fine sun vector				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Sun X	INT	Sun X. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Sun Y	INT	Sun Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Sun Z	INT	Sun Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 110: Nadir Vector Telemetry Format

ID	154		Frame Length (bytes)		6
Description	Measured nadir vector				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Nadir X	INT	Nadir X. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Nadir Y	INT	Nadir Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Nadir Z	INT	Nadir Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 111: Rate Sensor Rates Telemetry Format

ID	155		Frame Length (bytes)		6
Description	Rate sensor measurements				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	X Angular Rate	INT	X Angular Rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	16	16	Y Angular Rate	INT	Y Angular Rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	32	16	Z Angular Rate	INT	Z Angular Rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01

Table 112: Wheel Speed Telemetry Format

ID	156	Frame Length (bytes)	6
Description	Wheel speed measurement		

Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	X Wheel Speed	INT	X Wheel Speed. (Unit of measure is [rpm])
	16	16	Y Wheel Speed	INT	Y Wheel Speed. (Unit of measure is [rpm])
	32	16	Z Wheel Speed	INT	Z Wheel Speed. (Unit of measure is [rpm])

Table 113: Magnetorquer Command Telemetry Format

ID	157	Frame Length (bytes)	6	
Description	Magnetorquer commands			
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	X Magnetorquer Command	INT	X Magnetorquer Commanded on-time. (Unit of measure is [10ms units])
	16	16	Y Magnetorquer Command	INT	Y Magnetorquer Commanded on-time. (Unit of measure is [10ms units])
	32	16	Z Magnetorquer Command	INT	Z Magnetorquer Commanded on-time. (Unit of measure is [10ms units])

Table 114: Wheel Speed Commands Telemetry Format

ID	158	Frame Length (bytes)	6	
Description	Wheel speed commands			
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Commanded X Wheel Speed	INT	X Wheel Speed. (Unit of measure is [rpm])
	16	16	Commanded Y Wheel Speed	INT	Y Wheel Speed. (Unit of measure is [rpm])
	32	16	Commanded Z Wheel Speed	INT	Z Wheel Speed. (Unit of measure is [rpm])

Table 115: IGRF Modelled Magnetic Field Vector Telemetry Format

ID	159	Frame Length (bytes)	6

Description	IGRF modelled magnetic field vector (orbit frame referenced)				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	IGRF Modelled Magnetic Field X	INT	IGRF Modelled Magnetic Field X. Formatted value is obtained using the formula: (formatted value) [uT] = RAWVAL*0.01
	16	16	IGRF Modelled Magnetic Field Y	INT	IGRF Modelled Magnetic Field Y. Formatted value is obtained using the formula: (formatted value) [uT] = RAWVAL*0.01
	32	16	IGRF Modelled Magnetic Field Z	INT	IGRF Modelled Magnetic Field Z. Formatted value is obtained using the formula: (formatted value) [uT] = RAWVAL*0.01

Table 116: Modelled Sun Vector Telemetry Format

ID	160		Frame Length (bytes)		6
Description	Modelled sun vector (orbit frame referenced)				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Modelled Sun Vector X	INT	Modelled Sun Vector X. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Modelled Sun Vector Y	INT	Modelled Sun Vector Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Modelled Sun Vector Z	INT	Modelled Sun Vector Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 117: Estimated Gyro Bias Telemetry Format

ID	161		Frame Length (bytes)		6
Description	Estimated rate sensor bias				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description

	0	16	Estimated X-gyro Bias	INT	Estimated X-gyro Bias. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	16	16	Estimated Y-gyro Bias	INT	Estimated Y-gyro Bias. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	32	16	Estimated Z-gyro Bias	INT	Estimated Z-gyro Bias. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001

Table 118: Estimation Innovation Vector Telemetry Format

ID	162		**Frame Length (bytes)**		6
Description	Estimation innovation vector				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Innovation Vector X	INT	Innovation Vector X. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001
	16	16	Innovation Vector Y	INT	Innovation Vector Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001
	32	16	Innovation Vector Z	INT	Innovation Vector Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001

Table 119: Quaternion Error Vector Telemetry Format

ID	163		**Frame Length (bytes)**		6
Description	Quaternion error vector				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Quaternion Error – Q1	INT	Quaternion Error – Q1. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001

	16	16	Quaternion Error – Q2	INT	Quaternion Error – Q2. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001
	32	16	Quaternion Error – Q3	INT	Quaternion Error – Q3. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001

Table 120: Quaternion Covariance Telemetry Format

ID	164		**Frame Length (bytes)**		6
Description	Quaternion covariance				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Quaternion Covariance – Q1 RMS	INT	Quaternion Covariance – Q1 RMS. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001
	16	16	Quaternion Covariance – Q2 RMS	INT	Quaternion Covariance – Q2 RMS. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001
	32	16	Quaternion Covariance – Q3 RMS	INT	Quaternion Covariance – Q3 RMS. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001

Table 121: Angular Rate Covariance Telemetry Format

ID	165		**Frame Length (bytes)**		6
Description	Angular rate covariance				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	X Angular Rate Covariance	INT	X Angular Rate Covariance. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001
	16	16	Y Angular Rate Covariance	INT	Y Angular Rate Covariance. Formatted value is obtained

					using the formula: (formatted value) = RAWVAL*0.001
	32	16	Z Angular Rate Covariance	INT	Z Angular Rate Covariance. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001

Table 122: Raw Cam2 Sensor Telemetry Format

ID	166		**Frame Length (bytes)**		6
Description	Cam2 sensor capture and detection result				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Cam2 centroid X	INT	Cam2 azimuth angle
	16	16	Cam2 centroid Y	INT	Cam2 elevation angle
	32	8	Cam2 Capture status	ENUM	Cam2 capture status. Possible values are in Table 123: CaptureResult Enumeration Values
	40	8	Cam2 Detection result	ENUM	Cam2 detection result. Possible values are in Table 124: DetectResult Enumeration Values

Table 123: CaptureResult Enumeration Values

Numeric Value	Name	Description
0	Startup	Start-up
1	Pending	Capture pending
2	Success	Successfully captured
3	SuccessShift	Successfully captured on other SRAM (only applicable to CubeSense V2 hardware)
4	Timeout	Camera timeout
5	SRAMError	SRAM overcurrent

Table 124: DetectResult Enumeration Values

Numeric Value	Name	Description
0	Startup	Start-up
1	NoDetect	No detection scheduled
2	Pending	Detection pending
3	TooManyEdges	Nadir error – too many detected edges

4	TooFewEdges	Nadir error – not enough edges detected
5	BadFit	Nadir error – bad fit
6	SunNotFound	Sun error – sun not found
7	Success	Successful detection

Table 125: Raw Cam1 Sensor Telemetry Format

ID	167		**Frame Length (bytes)**		6
Description	Cam1 sensor capture and detection result				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Cam1 centroid X	INT	Cam1 azimuth angle
	16	16	Cam1 centroid Y	INT	Cam1 elevation angle
	32	8	Cam1 Capture status	ENUM	Cam1 capture status. Possible values are in Table 123: CaptureResult Enumeration Values
	40	8	Cam1 Detection result	ENUM	Cam1 detection result. Possible values are in Table 124: DetectResult Enumeration Values

Table 126: Raw CSS 1 to 6 Telemetry Format

ID	168		**Frame Length (bytes)**		6
Description	Raw CSS measurements 1 to 6				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	CSS1	UINT	sampled A/D value – corresponds to COS(sun_angle)
	8	8	CSS2	UINT	sampled A/D value – corresponds to COS(sun_angle)
	16	8	CSS3	UINT	sampled A/D value – corresponds to COS(sun_angle)
	24	8	CSS4	UINT	sampled A/D value – corresponds to COS(sun_angle)
	32	8	CSS5	UINT	sampled A/D value – corresponds to COS(sun_angle)
	40	8	CSS6	UINT	sampled A/D value – corresponds to COS(sun_angle)

Table 127: Raw CSS 7 to 10 Telemetry Format

ID	169		Frame Length (bytes)		6
Description	Raw CSS measurements 7 to 10				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	CSS7	UINT	sampled A/D value – corresponds to COS(sun_angle)
	8	8	CSS8	UINT	sampled A/D value – corresponds to COS(sun_angle)
	16	8	CSS9	UINT	sampled A/D value – corresponds to COS(sun_angle)
	24	8	CSS10	UINT	sampled A/D value – corresponds to COS(sun_angle)

Table 128: Raw Magnetometer Telemetry Format

ID	170		Frame Length (bytes)		6
Description	Raw magnetometer measurements				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	MagX	INT	sampled A/D value
	16	16	MagY	INT	sampled A/D value
	32	16	MagZ	INT	sampled A/D value

Table 129: CubeSense1 Current Measurements Telemetry Format

ID	171		Frame Length (bytes)		4
Description	CubeSense1 current measurements				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	CubeSense1 3V3 Current	UINT	CubeSense1 3V3 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1
	16	16	CubeSense1 Cam SRAM Current	UINT	CubeSense1 Cam SRAM Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1

Table 130: CubeControl Current Measurements Telemetry Format

ID	172		Frame Length (bytes)		6
Description	CubeControl current measurements				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	CubeControl 3V3 Current	UINT	CubeControl 3V3 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.48828125
	16	16	CubeControl 5V Current	UINT	CubeControl 5V Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.48828125
	32	16	CubeControl Vbat Current	UINT	CubeControl Vbat Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.48828125

Table 131: Wheel Currents Telemetry Format

ID	173		Frame Length (bytes)		6
Description	XYZ Wheel current measurement				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Wheel1Current	UINT	Wheel1 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.01
	16	16	Wheel2Current	UINT	Wheel2 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.01
	32	16	Wheel3Current	UINT	Wheel3 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.01

Table 132: ADCS Temperatures Telemetry Format

ID	174		Frame Length (bytes)		6
Description	Magnetometer + MCU temperature measurements				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	MCU Temperature	INT	MCU Temperature. (Unit of measure is [C])
	16	16	Magnetometer Temperature	INT	Magnetometer Temperature. Formatted value is obtained using the formula: (formatted value) [C] = RAWVAL/10.0
	32	16	Redundant Magnetometer Temperature	INT	Redundant Magnetometer Temperature. Formatted value is obtained using the formula: (formatted value) [C] = RAWVAL/10.0

Table 133: Rate sensor temperatures Telemetry Format

ID	175		Frame Length (bytes)		6
Description	Rate sensor temperatures				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	X-Rate Sensor Temperature	INT	X-Rate sensor Temperature. (Unit of measure is [C])
	16	16	Y-Rate Sensor Temperature	INT	Y-Rate sensor Temperature. (Unit of measure is [C])
	32	16	Z-Rate Sensor Temperature	INT	Z-Rate sensor Temperature. (Unit of measure is [C])

Table 134: Raw GPS Status Telemetry Format

ID	176		Frame Length (bytes)		6
Description	Raw GPS status				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	Gps Solution Status	ENUM	GPS Solution Status. Possible values are in Table 85: GpsSolutionStatus Enumeration Values

	8	8	Number of tracked GPS satellites	UINT	Number of tracked GPS satellites
	16	8	Number of GPS satellites used in solution	UINT	Number of GPS satellites used in solution
	24	8	Counter for XYZ Lof from GPS	UINT	Counter for XYZ Lof from GPS
	32	8	Counter for RANGE log from GPS	UINT	Counter for RANGE log from GPS
	40	8	Response Message for GPS log setup	UINT	Response Message for GPS log setup – p656 of OEMV615 reference manual

Table 135: Raw GPS Time Telemetry Format

ID	177		**Frame Length (bytes)**		6
Description	Raw GPS time				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	GPS Reference Week	UINT	GPS Reference Week
	16	32	GPS Time Milliseconds	UINT	GPS Time Milliseconds. (Unit of measure is [ms])

Table 136: Raw GPS X Telemetry Format

ID	178		**Frame Length (bytes)**		6
Description	Raw GPS X position and velocity (ECI referenced)				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	ECEF Position X	INT	ECEF Position X. (Unit of measure is [m])
	32	16	ECEF Velocity X	INT	ECEF Velocity X. (Unit of measure is [m/s])

Table 137: Raw GPS Y Telemetry Format

ID	179		**Frame Length (bytes)**		6
Description	Raw GPS Y position and velocity (ECI referenced)				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**

	0	32	ECEF Position Y	INT	ECEF Position Y. (Unit of measure is [m])
	32	16	ECEF Velocity Y	INT	ECEF Velocity Y. (Unit of measure is [m/s])

Table 138: Raw GPS Z Telemetry Format

ID	180		**Frame Length (bytes)**		6
Description	Raw GPS Z position and velocity (ECI referenced)				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	ECEF Position Z	INT	ECEF Position Z. (Unit of measure is [m])
	32	16	ECEF Velocity Z	INT	ECEF Velocity Z. (Unit of measure is [m/s])

Table 139: Star 1 Body Vector Telemetry Format

ID	181		**Frame Length (bytes)**		6
Description	Star 1 Body Vector				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Star1BX	INT	Star1 body X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Star1BY	INT	Star1 body Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Star1BZ	INT	Star1 body Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 140: Star 2 Body Vector Telemetry Format

ID	182		**Frame Length (bytes)**		6
Description	Star 2 Body Vector				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**

	0	16	Star2BX	INT	Star2 body X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Star2BY	INT	Star2 body Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Star2BZ	INT	Star2 body Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 141: Star 3 Body Vector Telemetry Format

ID	183		**Frame Length (bytes)**		6
Description	Star 3 Body Vector				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Star3BX	INT	Star3 body X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Star3BY	INT	Star3 body Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Star3BZ	INT	Star3 body Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 142: Star 1 Orbit Vector Telemetry Format

ID	184		**Frame Length (bytes)**		6
Description	Star 1 Orbit Vector				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Star1OX	INT	Star1 orbit X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

	16	16	Star1OY	INT	Star1 orbit Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Star1OZ	INT	Star1 orbit Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 143: Star 2 Orbit Vector Telemetry Format

ID	185		**Frame Length (bytes)**		6
Description	Star 2 Orbit Vector				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Star2OX	INT	Star2 orbit X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Star2OY	INT	Star2 orbit Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Star2OZ	INT	Star2 orbit Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 144: Star 3 Orbit Vector Telemetry Format

ID	186		**Frame Length (bytes)**		6
Description	Star 3 Orbit Vector				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Star3OX	INT	Star3 orbit X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Star3OY	INT	Star3 orbit Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

	32	16	Star3OZ	INT	Star3 orbit Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 145: Star Magnitude Telemetry Format

ID	187		**Frame Length (bytes)**		6
Description	Instrument magnitude of identified stars				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Magnitude Star 1	UINT	Instrument magnitude of star 1
	16	16	Magnitude Star 2	UINT	Instrument magnitude of star 2
	32	16	Magnitude Star 3	UINT	Instrument magnitude of star 3

Table 146: Star Performance1 Telemetry Format

ID	188		**Frame Length (bytes)**		6
Description	Performance parameters of star measurement				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Number of stars detected	UINT	Number of stars detected. (Unit of measure is [stars])
	8	8	Star image noise	UINT	Star image noise. (Unit of measure is [noise])
	16	8	Invalid Stars	UINT	Number of invalid stars detected. (Unit of measure is [Invalid stars])
	24	8	Number of stars identified	UINT	Number of stars identified. (Unit of measure is [stars])
	32	8	Identification mode	ENUM	Identification mode. Possible values are in Table 147: StarIDModeVal Enumeration Values
	40	8	Image dark value	UINT	The average value of center line in image. (Unit of measure is [8-bit pixel value])

Table 147: StarIDModeVal Enumeration Values

Numeric Value	Name	Description
0	Tracking Mode	

1	Lost Mode	

Table 148: Star Timing Telemetry Format

ID	189		Frame Length (bytes)		6
Description	Timing information of star measurement				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Capture	UINT	Capture. (Unit of measure is [ms])
	16	16	Detection	UINT	Detection. (Unit of measure is [ms])
	32	16	Identification	UINT	Identification. (Unit of measure is [ms])

Table 149: ADCS State Telemetry Format

ID	190		Frame Length (bytes)		54
Description	Current ADCS state				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	4	Attitude Estimation Mode	ENUM	Current attitude estimation mode. Possible values are in Table 80: EstimModeSelect Enumeration Values
	4	4	Control Mode	ENUM	Current attitude control mode. Possible values are in Table 78: ConModeSelect Enumeration Values
	8	2	ADCS Run Mode	ENUM	Current ADCS Running mode. Possible values are in Table 75: AdcsRunMode Enumeration Values
	10	2	ASGP4 Mode	ENUM	ASGP4 enabled state. Possible values are in Table 87: Asgp4ModeSelect Enumeration Values
	12	1	CubeControl Signal Enabled	BOOL	CubeControl Signal electronics enabled status
	13	1	CubeControl Motor Enabled	BOOL	CubeControl Motor electronics enabled status
	14	1	CubeSense1 Enabled	BOOL	CubeSense1 enabled status

	15	1	CubeSense2 Enabled	BOOL	CubeSense2 enabled status
	16	1	CubeWheel1 Enabled	BOOL	CubeWheel1 enabled status
	17	1	CubeWheel2 Enabled	BOOL	CubeWheel2 enabled status
	18	1	CubeWheel3 Enabled	BOOL	CubeWheel3 enabled status
	19	1	CubeStar Enabled	BOOL	CubeStar enabled status
	20	1	GPS Receiver Enabled	BOOL	GPS Receiver enabled status
	21	1	GPS LNA Power Enabled	BOOL	GPS Antenna LNA enabled status
	22	1	Motor Driver Enabled	BOOL	Motor Driver Electronics enabled status
	23	1	Sun is Above Local Horizon	BOOL	Sun is above the local horizon (elevation > 0)
	24	1	CubeSense1 Communications Error	BOOL	Communication error occurred with the CubeSense1
	25	1	CubeSense2 Communications Error	BOOL	Communication error occurred with the CubeSense2
	26	1	CubeControl Signal Communications Error	BOOL	Communication error occurred with the CubeControl Signal MCU
	27	1	CubeControl Motor Communications Error	BOOL	Communication error occurred with the CubeControl Motor MCU
	28	1	CubeWheel1 Communications Error	BOOL	Communication error occurred with the CubeWheel1
	29	1	CubeWheel2 Communications Error	BOOL	Communication error occurred with the CubeWheel2
	30	1	CubeWheel3 Communications Error	BOOL	Communication error occurred with the CubeWheel3
	31	1	CubeStar Communications Error	BOOL	Communication error occurred with the CubeStar

	32	1	Magnetometer Range Error	BOOL	Magnetometer measured magnetic field with size 65 uT
	33	1	Cam1 SRAM Overcurrent Detected	BOOL	Cam1 SRAM overcurrent detected
	34	1	Cam1 3V3 Overcurrent Detected	BOOL	Cam1 3V3 overcurrent detected
	35	1	Cam1 Sensor Busy Error	BOOL	Cam1 sensor was not idle at the start of ADCS loop
	36	1	Cam1 Sensor Detection Error	BOOL	Cam1 sensor was unable to compute angles (could be not in FOV)
	37	1	Sun Sensor Range Error	BOOL	Detected sun angles were outside of +/- 90 deg
	38	1	Cam2 SRAM Overcurrent Detected	BOOL	Cam2 SRAM overcurrent detected
	39	1	Cam2 3V3 Overcurrent Detected	BOOL	Cam2 3V3 overcurrent detected
	40	1	Cam2 Sensor Busy Error	BOOL	Cam2 sensor was not idle at the start of ADCS loop
	41	1	Cam2 Sensor Detection Error	BOOL	Cam2 sensor was unable to compute angles (could be not in FOV)
	42	1	Nadir Sensor Range Error	BOOL	Detected nadir angles were outside of +/- 60 deg
	43	1	Rate Sensor Range Error	BOOL	Measured XYZ-body rate is outside of the range +/-20 deg/s
	44	1	Wheel Speed Range Error	BOOL	Wheel XYZ speed measurement was outside the range +/-8500 rpm
	45	1	Coarse Sun Sensor Error	BOOL	Unable to compute Coarse Sun vector (could be not in FOV)
	46	1	StarTracker Match Error	BOOL	Unable to obtain enough matched stars
	47	1	Star Tracker Overcurrent Detected	BOOL	Star tracker overcurrent detected
	48	1	Orbit Parameters are Invalid	BOOL	Orbit Parameters are not in allowed bounds (angle

					exceeding limits etc.). Failed to initialize SGP4 propagator using supplied parameters
	49	1	Configuration is Invalid	BOOL	Magnetorquer Configuration or CSS in invalid. Each principle axis should have a torquer output (1,2, or 3) assigned. At least one CSS per principle axis needed
	50	1	Control Mode Change is not allowed	BOOL	Attempt was made to select control mode without appropriate estimator, or command to set Y-momentum mode while not in steady-state Y-Thomson
	51	1	Estimator Change is not allowed	BOOL	Attempt was made to change to an estimation mode that would be inappropriate for the current control mode
	52	2	Current Magnetometer Sampling Mode	ENUM	Current magnetometer sampling mode. Possible values are in Table 90: MagModeVal Enumeration Values
	54	1	Modelled and measured magnetic field differs in size	BOOL	Modelled and measured magnetic field differs in size by more than 5000 nT
	55	1	Node Recovery Error	BOOL	Failed to Recover an ADCS Node by successive resets
	56	1	CubeSense1 Runtime Error	BOOL	Runtime error occurred with the CubeSense1
	57	1	CubeSense2 Runtime Error	BOOL	Runtime error occurred with the CubeSense2
	58	1	CubeControl Signal Runtime Error	BOOL	Runtime error occurred with the CubeControl Signal MCU
	59	1	CubeControl Motor Runtime Error	BOOL	Runtime error occurred with the CubeControl Motor MCU
	60	1	CubeWheel1 Runtime Error	BOOL	Runtime error occurred with the CubeWheel1
	61	1	CubeWheel2 Runtime Error	BOOL	Runtime error occurred with the CubeWheel2
	62	1	CubeWheel3 Runtime Error	BOOL	Runtime error occurred with the CubeWheel3

	63	1	CubeStar Runtime Error	BOOL	Runtime error occurred with the CubeStar
	64	1	Magnetometer Error	BOOL	Magnetometer failure occurred
	65	1	Rate Sensor Failure	BOOL	Rate sensor failure occurred
	96	16	Estimated Roll Angle	INT	Estimated roll angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	112	16	Estimated Pitch Angle	INT	Estimated pitch angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	128	16	Estimated Yaw Angle	INT	Estimated yaw angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	144	16	Estimated q1	INT	Estimated q1
	160	16	Estimated q2	INT	Estimated q2
	176	16	Estimated q3	INT	Estimated q3
	192	16	Estimated X Angular Rate	INT	Estimated X angular rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	208	16	Estimated Y Angular Rate	INT	Estimated Y angular rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	224	16	Estimated Z Angular Rate	INT	Estimated Z angular rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	240	16	X position	INT	ECI referenced X coordinate. Formatted value is obtained using the formula: (formatted value) [km] = RAWVAL*0.25
	256	16	Y position	INT	ECI referenced Y coordinate. Formatted value is obtained using the formula: (formatted value) [km] = RAWVAL*0.25
	272	16	Z position	INT	ECI referenced Z coordinate. Formatted value is obtained

					using the formula: (formatted value) [km] = RAWVAL*0.25
	288	16	X Velocity	INT	ECI referenced X velocity. Formatted value is obtained using the formula: (formatted value) [m/s] = RAWVAL*0.25
	304	16	Y Velocity	INT	ECI referenced Y velocity. Formatted value is obtained using the formula: (formatted value) [m/s] = RAWVAL*0.25
	320	16	Z Velocity	INT	ECI referenced Z velocity. Formatted value is obtained using the formula: (formatted value) [m/s] = RAWVAL*0.25
	336	16	Latitude	INT	WGS-84 Latitude angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	352	16	Longitude	INT	Longitude angle. Formatted value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	368	16	Altitude	UINT	WGS-84 altitude. Formatted value is obtained using the formula: (formatted value) [km] = RAWVAL*0.01
	384	16	ECEF Position X	INT	ECEF Position X. (Unit of measure is [m])
	400	16	ECEF Position Y	INT	ECEF Position Y. (Unit of measure is [m])
	416	16	ECEF Position Z	INT	ECEF Position Z. (Unit of measure is [m])

Table 150: ADCS Measurements Telemetry Format

ID	191		**Frame Length (bytes)**		72
Description	Calibrated sensor measurements				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Magnetic Field X	INT	Magnetic Field X. Formatted value is obtained using the

					formula: (formatted value) [uT] = RAWVAL*0.01
	16	16	Magnetic Field Y	INT	Magnetic Field Y. Formatted value is obtained using the formula: (formatted value) [uT] = RAWVAL*0.01
	32	16	Magnetic Field Z	INT	Magnetic Field Z. Formatted value is obtained using the formula: (formatted value) [uT] = RAWVAL*0.01
	48	16	Coarse Sun X	INT	Coarse Sun X. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	64	16	Coarse Sun Y	INT	Coarse Sun Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	80	16	Coarse Sun Z	INT	Coarse Sun Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	96	16	Sun X	INT	Sun X. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	112	16	Sun Y	INT	Sun Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	128	16	Sun Z	INT	Sun Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	144	16	Nadir X	INT	Nadir X. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	160	16	Nadir Y	INT	Nadir Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

	176	16	Nadir Z	INT	Nadir Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	192	16	X Angular Rate	INT	X Angular Rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	208	16	Y Angular Rate	INT	Y Angular Rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	224	16	Z Angular Rate	INT	Z Angular Rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.01
	240	16	X Wheel Speed	INT	X Wheel Speed. (Unit of measure is [rpm])
	256	16	Y Wheel Speed	INT	Y Wheel Speed. (Unit of measure is [rpm])
	272	16	Z Wheel Speed	INT	Z Wheel Speed. (Unit of measure is [rpm])
	288	16	Star1BX	INT	Star1 body X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	304	16	Star1BY	INT	Star1 body Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	320	16	Star1BZ	INT	Star1 body Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	336	16	Star1OX	INT	Star1 orbit X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	352	16	Star1OY	INT	Star1 orbit Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

	368	16	Star1OZ	INT	Star1 orbit Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	384	16	Star2BX	INT	Star2 body X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	400	16	Star2BY	INT	Star2 body Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	416	16	Star2BZ	INT	Star2 body Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	432	16	Star2OX	INT	Star2 orbit X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	448	16	Star2OY	INT	Star2 orbit Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	464	16	Star2OZ	INT	Star2 orbit Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	480	16	Star3BX	INT	Star3 body X-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	496	16	Star3BY	INT	Star3 body Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	512	16	Star3BZ	INT	Star3 body Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	528	16	Star3OX	INT	Star3 orbit X-vector. Formatted value is obtained using the

					formula: (formatted value) = RAWVAL/10000.0
	544	16	Star3OY	INT	Star3 orbit Y-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	560	16	Star3OZ	INT	Star3 orbit Z-vector. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 151: Actuator Commands Telemetry Format

ID	192		**Frame Length (bytes)**		12
Description	Actuator commands				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	X Magnetorquer Command	INT	X Magnetorquer Commanded on-time. (Unit of measure is [10ms units])
	16	16	Y Magnetorquer Command	INT	Y Magnetorquer Commanded on-time. (Unit of measure is [10ms units])
	32	16	Z Magnetorquer Command	INT	Z Magnetorquer Commanded on-time. (Unit of measure is [10ms units])
	48	16	Commanded X Wheel Speed	INT	X Wheel Speed. (Unit of measure is [rpm])
	64	16	Commanded Y Wheel Speed	INT	Y Wheel Speed. (Unit of measure is [rpm])
	80	16	Commanded Z Wheel Speed	INT	Z Wheel Speed. (Unit of measure is [rpm])

Table 152: Estimation Data Telemetry Format

ID	193		**Frame Length (bytes)**		42
Description	Estimation meta-data				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	IGRF Modelled Magnetic Field X	INT	IGRF Modelled Magnetic Field X. Formatted value is obtained

					using the formula: (formatted value) [uT] = RAWVAL*0.01
	16	16	IGRF Modelled Magnetic Field Y	INT	IGRF Modelled Magnetic Field Y. Formatted value is obtained using the formula: (formatted value) [uT] = RAWVAL*0.01
	32	16	IGRF Modelled Magnetic Field Z	INT	IGRF Modelled Magnetic Field Z. Formatted value is obtained using the formula: (formatted value) [uT] = RAWVAL*0.01
	48	16	Modelled Sun Vector X	INT	Modelled Sun Vector X. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	64	16	Modelled Sun Vector Y	INT	Modelled Sun Vector Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	80	16	Modelled Sun Vector Z	INT	Modelled Sun Vector Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	96	16	Estimated X-gyro Bias	INT	Estimated X-gyro Bias. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	112	16	Estimated Y-gyro Bias	INT	Estimated Y-gyro Bias. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	128	16	Estimated Z-gyro Bias	INT	Estimated Z-gyro Bias. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	144	16	Innovation Vector X	INT	Innovation Vector X. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001
	160	16	Innovation Vector Y	INT	Innovation Vector Y. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001

	176	16	Innovation Vector Z	INT	Innovation Vector Z. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001
	192	16	Quaternion Error – Q1	INT	Quaternion Error – Q1. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001
	208	16	Quaternion Error – Q2	INT	Quaternion Error – Q2. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001
	224	16	Quaternion Error – Q3	INT	Quaternion Error – Q3. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.0001
	240	16	Quaternion Covariance – Q1 RMS	INT	Quaternion Covariance – Q1 RMS. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001
	256	16	Quaternion Covariance – Q2 RMS	INT	Quaternion Covariance – Q2 RMS. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001
	272	16	Quaternion Covariance – Q3 RMS	INT	Quaternion Covariance – Q3 RMS. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001
	288	16	X Angular Rate Covariance	INT	X Angular Rate Covariance. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001
	304	16	Y Angular Rate Covariance	INT	Y Angular Rate Covariance. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001
	320	16	Z Angular Rate Covariance	INT	Z Angular Rate Covariance. Formatted value is obtained using the formula: (formatted value) = RAWVAL*0.001

Table 153: Raw Sensor Measurements Telemetry Format

ID	194		Frame Length (bytes)		34
Description	Raw sensor measurements				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Cam2 centroid X	INT	Cam2 azimuth angle
	16	16	Cam2 centroid Y	INT	Cam2 elevation angle
	32	8	Cam2 Capture status	ENUM	Cam2 capture status. Possible values are in Table 123: CaptureResult Enumeration Values
	40	8	Cam2 Detection result	ENUM	Cam2 detection result. Possible values are in Table 124: DetectResult Enumeration Values
	48	16	Cam1 centroid X	INT	Cam1 azimuth angle
	64	16	Cam1 centroid Y	INT	Cam1 elevation angle
	80	8	Cam1 Capture status	ENUM	Cam1 capture status. Possible values are in Table 123: CaptureResult Enumeration Values
	88	8	Cam1 Detection result	ENUM	Cam1 detection result. Possible values are in Table 124: DetectResult Enumeration Values
	96	8	CSS1	UINT	sampled A/D value – corresponds to COS(sun_angle)
	104	8	CSS2	UINT	sampled A/D value – corresponds to COS(sun_angle)
	112	8	CSS3	UINT	sampled A/D value – corresponds to COS(sun_angle)
	120	8	CSS4	UINT	sampled A/D value – corresponds to COS(sun_angle)
	128	8	CSS5	UINT	sampled A/D value – corresponds to COS(sun_angle)
	136	8	CSS6	UINT	sampled A/D value – corresponds to COS(sun_angle)
	144	8	CSS7	UINT	sampled A/D value – corresponds to COS(sun_angle)
	152	8	CSS8	UINT	sampled A/D value – corresponds to COS(sun_angle)
	160	8	CSS9	UINT	sampled A/D value – corresponds to COS(sun_angle)

	168	8	CSS10	UINT	sampled A/D value – corresponds to COS(sun_angle)
	176	16	MagX	INT	sampled A/D value
	192	16	MagY	INT	sampled A/D value
	208	16	MagZ	INT	sampled A/D value
	224	16	RateX	INT	sampled A/D value
	240	16	RateY	INT	sampled A/D value
	256	16	RateZ	INT	sampled A/D value

Table 154: Power and Temperature Measurements Telemetry Format

ID	195		**Frame Length (bytes)**		38
Description	Power and temperature measurements				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	CubeSense1 3V3 Current	UINT	CubeSense1 3V3 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1
	16	16	CubeSense1 Cam SRAM Current	UINT	CubeSense1 Cam SRAM Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1
	32	16	CubeSense2 3V3 Current	UINT	CubeSense2 3V3 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1
	48	16	CubeSense2 Cam SRAM Current	UINT	CubeSense2 Cam SRAM Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1
	64	16	CubeControl 3V3 Current	UINT	CubeControl 3V3 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.48828125
	80	16	CubeControl 5V Current	UINT	CubeControl 5V Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.48828125

	96	16	CubeControl Vbat Current	UINT	CubeControl Vbat Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.48828125
	112	16	Wheel1Current	UINT	Wheel1 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.01
	128	16	Wheel2Current	UINT	Wheel2 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.01
	144	16	Wheel3Current	UINT	Wheel3 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.01
	160	16	CubeStarCurrent	UINT	CubeStar Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.01
	176	16	Magnetorquer Current	UINT	Magnetorquer Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1
	192	16	CubeStar MCU temperature	INT	CubeStar MCU temperature. Formatted value is obtained using the formula: (formatted value) [C] = RAWVAL/100.0
	208	16	MCU Temperature	INT	MCU Temperature. (Unit of measure is [C])
	224	16	Magnetometer Temperature	INT	Magnetometer Temperature. Formatted value is obtained using the formula: (formatted value) [C] = RAWVAL/10.0
	240	16	Redundant Magnetometer Temperature	INT	Redundant Magnetometer Temperature. Formatted value is obtained using the formula: (formatted value) [C] = RAWVAL/10.0
	256	16	X-Rate Sensor Temperature	INT	X-Rate sensor Temperature. (Unit of measure is [C])
	272	16	Y-Rate Sensor Temperature	INT	Y-Rate sensor Temperature. (Unit of measure is [C])

	288	16	Z-Rate Sensor Temperature	INT	Z-Rate sensor Temperature. (Unit of measure is [C])

Table 155: Adcs Execution Times Telemetry Format

ID	196		**Frame Length (bytes)**		8
Description	Returns information about execution times of ACP functions				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Time to Perform ADCS Update	UINT	Time to perform complete ADCS Update function. (Unit of measure is [ms])
	16	16	Time to Perform Sensor/Actuator Communications	UINT	Time to perform Sensor/actuator communications. (Unit of measure is [ms])
	32	16	Time to Execute SGP4 Propagator	UINT	Time to execute SGP4 propagator. (Unit of measure is [ms])
	48	16	Time to Execute IGRF Model	UINT	Time to execute IGRF computation. (Unit of measure is [ms])

Table 156: ADCS Misc Current Measurements Telemetry Format

ID	198		**Frame Length (bytes)**		6
Description	CubeStar and Torquer current and temperature measurements				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	CubeStarCurrent	UINT	CubeStar Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.01
	16	16	Magnetorquer Current	UINT	Magnetorquer Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1
	32	16	CubeStar MCU temperature	INT	CubeStar MCU temperature. Formatted value is obtained using the formula: (formatted value) [C] = RAWVAL/100.0

Table 157: Fine Estimated Angular Rates Telemetry Format

ID	201		Frame Length (bytes)		6
Description	High resolution estimated angular rates relative to orbit reference frame				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Estimated X Angular Rate	INT	Estimated X angular rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	16	16	Estimated Y Angular Rate	INT	Estimated Y angular rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	32	16	Estimated Z Angular Rate	INT	Estimated Z angular rate. Formatted value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001

Table 158: Raw GPS Measurements Telemetry Format

ID	210		Frame Length (bytes)		36
Description	Raw GPS measurements				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	Gps Solution Status	ENUM	GPS Solution Status. Possible values are in Table 85: GpsSolutionStatus Enumeration Values
	8	8	Number of tracked GPS satellites	UINT	Number of tracked GPS satellites
	16	8	Number of GPS satellites used in solution	UINT	Number of GPS satellites used in solution
	24	8	Counter for XYZ Lof from GPS	UINT	Counter for XYZ Lof from GPS
	32	8	Counter for RANGE log from GPS	UINT	Counter for RANGE log from GPS
	40	8	Response Message for GPS log setup	UINT	Response Message for GPS log setup – p656 of OEMV615 reference manual

	48	16	GPS Reference Week	UINT	GPS Reference Week
	64	32	GPS Time Milliseconds	UINT	GPS Time Milliseconds. (Unit of measure is [ms])
	96	32	ECEF Position X	INT	ECEF Position X. (Unit of measure is [m])
	128	16	ECEF Velocity X	INT	ECEF Velocity X. (Unit of measure is [m/s])
	144	32	ECEF Position Y	INT	ECEF Position Y. (Unit of measure is [m])
	176	16	ECEF Velocity Y	INT	ECEF Velocity Y. (Unit of measure is [m/s])
	192	32	ECEF Position Z	INT	ECEF Position Z. (Unit of measure is [m])
	224	16	ECEF Velocity Z	INT	ECEF Velocity Z. (Unit of measure is [m/s])
	240	8	X-pos Standard Deviation	UINT	X-pos Standard Deviation. Formatted value is obtained using the formula: (formatted value) [m] = RAWVAL*0.1
	248	8	Y-pos Standard Deviation	UINT	Y-pos Standard Deviation. Formatted value is obtained using the formula: (formatted value) [m] = RAWVAL*0.1
	256	8	Z-pos Standard Deviation	UINT	Z-pos Standard Deviation. Formatted value is obtained using the formula: (formatted value) [m] = RAWVAL*0.1
	264	8	X-vel Standard Deviation	UINT	X-vel Standard Deviation. (Unit of measure is [m/s])
	272	8	Y-vel Standard Deviation	UINT	Y-vel Standard Deviation. (Unit of measure is [m/s])
	280	8	Z-vel Standard Deviation	UINT	Z-vel Standard Deviation. (Unit of measure is [m/s])

Table 159: Raw Star Tracker Telemetry Format

ID	211		**Frame Length (bytes)**		54
Description	Raw Star Tracker Measurement				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**

	0	8	Number of stars detected	UINT	Number of stars detected. (Unit of measure is [stars])
	8	8	Star image noise	UINT	Star image noise. (Unit of measure is [noise])
	16	8	Invalid Stars	UINT	Number of invalid stars detected. (Unit of measure is [Invalid stars])
	24	8	Number of stars identified	UINT	Number of stars identified. (Unit of measure is [stars])
	32	8	Identification mode	ENUM	Identification mode. Possible values are in Table 147: StarIDModeVal Enumeration Values
	40	8	Image dark value	UINT	The average value of center line in image. (Unit of measure is [8-bit pixel value])
	48	1	Image Capture Success	BOOL	Image Capture Success
	49	1	Detection Success	BOOL	Detection Success
	50	1	Identification Success	BOOL	Identification Success
	51	1	AttitudeSuccess	BOOL	Attitude Success
	52	1	Processing Time Error	BOOL	Processing time Error
	53	1	Tracking Module Enabled	BOOL	Tracking Module Enabled
	54	1	Prediction Enabled	BOOL	Prediction Enabled
	55	1	Comms error	BOOL	Comms error
	56	16	Sample Period	UINT	Sample Period
	72	8	Star 1 confidence	UINT	Star 1 confidence. (Unit of measure is [percentage])
	80	8	Star 2 confidence	UINT	Star 2 confidence. (Unit of measure is [percentage])
	88	8	Star 3 confidence	UINT	Star 3 confidence. (Unit of measure is [percentage])
	96	16	Magnitude Star 1	UINT	Instrument magnitude of star 1
	112	16	Magnitude Star 2	UINT	Instrument magnitude of star 2
	128	16	Magnitude Star 3	UINT	Instrument magnitude of star 3
	144	16	Catalogue Star 1	UINT	Catalogue number of star 1
	160	16	Centroid X Star 1	INT	X centroid of star 1
	176	16	Centroid Y Star 1	INT	Y centroid of star 1
	192	16	Catalogue Star 2	UINT	Catalogue number of star 2
	208	16	Centroid X Star 2	INT	X centroid of star 2

	224	16	Centroid Y Star 2	INT	Y centroid of star 2
	240	16	Catalogue Star 3	UINT	Catalogue number of star 3
	256	16	Centroid X Star 3	INT	X centroid of star 3
	272	16	Centroid Y Star 3	INT	Y centroid of star 3
	288	16	Capture	UINT	Capture. (Unit of measure is [ms])
	304	16	Detection	UINT	Detection. (Unit of measure is [ms])
	320	16	Identification	UINT	Identification. (Unit of measure is [ms])
	336	16	x-Axis rate	INT	Estimated Rate around CubeStar X-axis. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	352	16	y-Axis rate	INT	Estimated Rate around CubeStar Y-axis. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	368	16	z-Axis rate	INT	Estimated Rate around CubeStar Z-axis. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	384	16	Q0	INT	CubeStar estimated attitude Q1. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	400	16	Q1	INT	CubeStar estimated attitude Q2. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	416	16	Q2	INT	CubeStar estimated attitude Q3. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 160: Star 1 Raw Data Telemetry Format

ID	212	**Frame Length (bytes)**	6
Description	Catalogue index and detected coordinates for star 1		

Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Catalogue Star 1	UINT	Catalogue number of star 1
	16	16	Centroid X Star 1	INT	X centroid of star 1
	32	16	Centroid Y Star 1	INT	Y centroid of star 1

Table 161: Star 2 Raw Data Telemetry Format

ID	213		Frame Length (bytes)		6
Description	Catalogue index and detected coordinates for star 2				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Catalogue Star 2	UINT	Catalogue number of star 2
	16	16	Centroid X Star 2	INT	X centroid of star 2
	32	16	Centroid Y Star 2	INT	Y centroid of star 2

Table 162: Star 3 Raw Data Telemetry Format

ID	214		Frame Length (bytes)		6
Description	Catalogue index and detected coordinates for star 3				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Catalogue Star 3	UINT	Catalogue number of star 3
	16	16	Centroid X Star 3	INT	X centroid of star 3
	32	16	Centroid Y Star 3	INT	Y centroid of star 3

Table 163: Secondary Magnetometer Raw Measurements Telemetry Format

ID	215		Frame Length (bytes)		6
Description	Secondary Magnetometer raw measurements				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	MagX	INT	sampled A/D value
	16	16	MagY	INT	sampled A/D value
	32	16	MagZ	INT	sampled A/D value

Table 164: Raw Rate Sensor Telemetry Format

ID	216	Frame Length (bytes)	6
Description	Raw rate sensor measurements		

Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	RateX	INT	sampled A/D value
	16	16	RateY	INT	sampled A/D value
	32	16	RateZ	INT	sampled A/D value

Table 165: Estimated Quaternion Telemetry Format

ID	218		Frame Length (bytes)		6
Description	Estimated quaternion set				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Estimated q1	INT	Estimated q1
	16	16	Estimated q2	INT	Estimated q2
	32	16	Estimated q3	INT	Estimated q3

Table 166: ECEF Position Telemetry Format

ID	219		Frame Length (bytes)		6
Description	Satellite position in ECEF coordinates				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	ECEF Position X	INT	ECEF Position X. (Unit of measure is [m])
	16	16	ECEF Position Y	INT	ECEF Position Y. (Unit of measure is [m])
	32	16	ECEF Position Z	INT	ECEF Position Z. (Unit of measure is [m])

Table 167: ACP Execution State Telemetry Format

ID	220		Frame Length (bytes)		3
Description	Returns information about the ACP loop				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Time Since Iteration Start	UINT	Time since the start of the current loop iteration. (Unit of measure is [ms])
	16	8	Current Execution Point	ENUM	Indicates which part of the loop is currently executing. Possible values are in Table 168:

					ExecutionWaypoints Enumeration Values

Table 168: ExecutionWaypoints Enumeration Values

Numeric Value	Name	Description
0	Init	Busy with initialization
1	Idle	Idle
2	Sensor/Actuator Communications	Sensor/Actuator Communications
3	ADCS Update	ADCS Estimation & Control Update
4	Peripheral Power commands (over I2C)	Peripheral Power commands (over I2C)
5	CPU Temperature Sampling	CPU Temperature Sampling
6	Image Download	Image Download
7	Image Compression	Image Compression
8	Saving Image to SD Card	Saving Image to SD Card
9	Logging	Logging
10	Log File Compression	Log File Compression
11	Saving Log to SD Card	Saving Log to SD Card
12	Writing to flash	Writing to flash memory

Table 169: Current ADCS State 2 Telemetry Format

ID	224		**Frame Length (bytes)**		6
Description	Current state of the Attitude Control Processor – frame 2				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	1	Orbit Parameters are Invalid	BOOL	Orbit Parameters are not in allowed bounds (angle exceeding limits etc.). Failed to initialize SGP4 propagator using supplied parameters
	1	1	Configuration is Invalid	BOOL	Magnetorquer Configuration or CSS in invalid. Each principle axis should have a torquer output (1,2, or 3) assigned. At least one CSS per principle axis needed
	2	1	Control Mode Change is not allowed	BOOL	Attempt was made to select control mode without appropriate estimator, or command to set Y-momentum

					mode while not in steady-state Y-Thomson
	3	1	Estimator Change is not allowed	BOOL	Attempt was made to change to an estimation mode that would be inappropriate for the current control mode
	4	2	Current Magnetometer Sampling Mode	ENUM	Current magnetometer sampling mode. Possible values are in Table 90: MagModeVal Enumeration Values
	6	1	Modelled and measured magnetic field differs in size	BOOL	Modelled and measured magnetic field differs in size by more than 5000 nT
	7	1	Node Recovery Error	BOOL	Failed to Recover an ADCS Node by successive resets
	8	1	CubeSense1 Runtime Error	BOOL	Runtime error occurred with the CubeSense1
	9	1	CubeSense2 Runtime Error	BOOL	Runtime error occurred with the CubeSense2
	10	1	CubeControl Signal Runtime Error	BOOL	Runtime error occurred with the CubeControl Signal MCU
	11	1	CubeControl Motor Runtime Error	BOOL	Runtime error occurred with the CubeControl Motor MCU
	12	1	CubeWheel1 Runtime Error	BOOL	Runtime error occurred with the CubeWheel1
	13	1	CubeWheel2 Runtime Error	BOOL	Runtime error occurred with the CubeWheel2
	14	1	CubeWheel3 Runtime Error	BOOL	Runtime error occurred with the CubeWheel3
	15	1	CubeStar Runtime Error	BOOL	Runtime error occurred with the CubeStar
	16	1	Magnetometer Error	BOOL	Magnetometer failure occurred
	17	1	Rate Sensor Failure	BOOL	Rate sensor failure occurred

Table 170: ASGP4 TLEs Telemetry Format

ID	228		**Frame Length (bytes)**		33
Description	ASGP4 TLEs generated				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	1	ASGP4 complete	BOOL	Is ASGP4 process complete

	1	7	ASGP4 error	ENUM	The error state that the asgp4 module is in. Possible values are in Table 171: AsgpError Enumeration Values
	8	32	ASGP4 Epoch	FLOAT	Epoch from asgp4
	40	32	ASGP4 inclination	FLOAT	Inclination from asgp4
	72	32	ASGP4 RAAN	FLOAT	RAAN from asgp4
	104	32	ASGP4 ECC	FLOAT	Eccentricity from asgp4
	136	32	ASGP4 AOP	FLOAT	AOP from asgp4
	168	32	ASGP4 MA	FLOAT	MA from asgp4
	200	32	ASGP4 MM	FLOAT	MM from asgp4
	232	32	ASGP4 Bstar	FLOAT	Bstar from asgp4

Table 171: AsgpError Enumeration Values

Numeric Value	Name	Description
0	Off	No error
1	Unix Time	Error due to time
2	Position	Error due to position error exceeding threshold
3	Overflow	Error overflow

Table 172: CubeStar Estimated Rates Telemetry Format

ID	229		Frame Length (bytes)		6
Description	Angular rates estimated by CubeStar				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	x-Axis rate	INT	Estimated Rate around CubeStar X-axis. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	y-Axis rate	INT	Estimated Rate around CubeStar Y-axis. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	z-Axis rate	INT	Estimated Rate around CubeStar Z-axis. Formatted value is obtained using the formula:

					(formatted value) = RAWVAL/10000.0

Table 173: CubeStar Estimated Quaternion Telemetry Format

ID	230		Frame Length (bytes)		6
Description	Attitude quaternion estimated by CubeStar				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Q0	INT	CubeStar estimated attitude Q1. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Q1	INT	CubeStar estimated attitude Q2. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Q2	INT	CubeStar estimated attitude Q3. Formatted value is obtained using the formula: (formatted value) = RAWVAL/10000.0

Table 174: Star Performance2 Telemetry Format

ID	231		Frame Length (bytes)		6
Description	Performance parameters of star measurement				
Channels	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	1	Image Capture Success	BOOL	Image Capture Success
	1	1	Detection Success	BOOL	Detection Success
	2	1	Identification Success	BOOL	Identification Success
	3	1	AttitudeSuccess	BOOL	Attitude Success
	4	1	Processing Time Error	BOOL	Processing time Error
	5	1	Tracking Module Enabled	BOOL	Tracking Module Enabled
	6	1	Prediction Enabled	BOOL	Prediction Enabled
	7	1	Comms error	BOOL	Comms error
	8	16	Sample Period	UINT	Sample Period

	24	8	Star 1 confidence	UINT	Star 1 confidence. (Unit of measure is [percentage])
	32	8	Star 2 confidence	UINT	Star 2 confidence. (Unit of measure is [percentage])
	40	8	Star 3 confidence	UINT	Star 3 confidence. (Unit of measure is [percentage])

Table 175: CubeSense2 Current Measurements Telemetry Format

ID	232		**Frame Length (bytes)**		4
Description	CubeSense2 current measurements				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	CubeSense2 3V3 Current	UINT	CubeSense2 3V3 Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1
	16	16	CubeSense2 Cam SRAM Current	UINT	CubeSense2 Cam SRAM Current. Formatted value is obtained using the formula: (formatted value) [mA] = RAWVAL*0.1

Table 176: Status of Image Capture and Save Operation Telemetry Format

ID	233		**Frame Length (bytes)**		2
Description	Status of Image Capture and Save Operation				
Channels	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Percentage Complete	UINT	Current progress of operation. (Unit of measure is [%])
	8	8	Status	ENUM	Current status of operation. Possible values are in Table 177: ImSaveStatus Enumeration Values

Table 177: ImSaveStatus Enumeration Values

Numeric Value	Name	Description
0	No Error	No Error
1	Timeout waiting for sensor to become available	Timeout waiting for sensor to become available

2	Timeout waiting for next frame to become ready	Timeout waiting for next frame to become ready
3	Checksum mismatch between downloaded frame and unit frame	Checksum mismatch between downloaded frame and unit frame
4	Error writing to SD card	Error writing to SD card

5.3.3 CubeAcp Configuration Messages

Table 178: List of CubeAcp Configuration Messages

Set ID	Get ID	Name	Description	Length (bytes)
Configuration				
21	136	Set Magnetorquer Configuration	Set magnetorquer configuration parameters – Table 179: Set Magnetorquer Configuration Message Format	3
22	137	Set Wheel Configuration	Set wheel configuration parameters – Table 181: Set Wheel Configuration Message Format	4
23	138	Set Rate Gyro Configuration	Set rate gyro configuration parameters – Table 182: Set Rate Gyro Configuration Message Format	10
24	139	CSS Configuration	Photodiode pointing directions and scale factors – Table 183: CSS Configuration Message Format	21
37	202	Set Star Tracker Configuration	Set configurations of CubeStar – Table 188: Set Star Tracker Configuration Message Format	53
25	203	CubeSense Configuration	CubeSense configuration parameters – Table 189: CubeSense Configuration Message Format	112
26	204	Magnetometer Configuration	Magnetometer configuration parameters – Table 190: Magnetometer Configuration Message Format	30
36	205	Redundant Magnetometer Configuration	Redundant magnetometer configuration parameters – Table 191: Redundant Magnetometer Configuration Message Format	30
20	206	ADCS Configuration	Current configuration – Table 192: ADCS Configuration Message Format	504
45	207	SGP4 Orbit Parameters	SGP4 Orbit Parameters – Table 194: SGP4 Orbit Parameters Message Format	64
38	208	Set Detumbling Control Parameters	Set controller gains and reference values for Detumbling control mode – Table 195: Set Detumbling Control Parameters Message Format	14

39	209	Set Y-Wheel Control Parameters	Set controller gains and reference value for Y-wheel control mode – Table 196: Set Y-Wheel Control Parameters Message Format	20
40	217	Set Reaction Wheel Control Parameters	Set controller gains and reference value for reaction wheel control mode – Table 197: Set Reaction Wheel Control Parameters Message Format	13
54	221	Set Tracking Controller Gain Parameters	Set controller gains for tracking control mode – Table 198: Set Tracking Controller Gain Parameters Message Format	13
41	222	Moment of Inertia Matrix	Satellite moment of inertia matrix – Table 199: Moment of Inertia Matrix Message Format	24
27	223	Estimation Parameters	Estimation noise covariance and sensor mask – Table 200: Estimation Parameters Message Format	31
30	225	ADCS System Configuration	Current hard-coded system configuration – Table 201: ADCS System Configuration Message Format	173
29	226	User-coded Controller and Estimator Parameters	Settings for user-coded estimation and control modes – Table 208: User-coded Controller and Estimator Parameters Message Format	96
28	227	Augmented-SGP4 Parameters	Settings for GPS augmented SGP4 – Table 209: Augmented-SGP4 Parameters Message Format	30
General				
11	197	ADCS Power Control	Control power to selected components – Table 184: ADCS Power Control Message Format	3
15	199	Commanded Attitude Angles	Commanded attitude angles – Table 186: Commanded Attitude Angles Message Format	6
55	200	Tracking Controller Target Reference	Target reference for tracking control mode – Table 187: Tracking Controller Target Reference Message Format	12
104	235	SD Log1 Configuration	Log selection and period for LOG1 – Table 210: SD Log1 Configuration Message Format	13
105	236	SD Log2 Configuration	Log selection and period for LOG2 – Table 212: SD Log2 Configuration Message Format	13
106	237	UART Log Configuration	Log selection and period for UART (unsolicited TLM) – Table 213: UART Log Configuration Message Format	12
34	238	Inertial Pointing Reference Vector	Reference unit vector for inertial pointing control mode – Table 214: Inertial Pointing Reference Vector Message Format	6

Table 179: Set Magnetorquer Configuration Message Format

Set ID/Get ID	21/136		Parameters Length (bytes)	3
Description	Set magnetorquer configuration parameters			
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	Magnetorquer 1 Configuration	ENUM	Magnetorquer 1 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	8	8	Magnetorquer 2 Configuration	ENUM	Magnetorquer 2 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	16	8	Magnetorquer 3 Configuration	ENUM	Magnetorquer 3 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values

Table 180: AxisSelect Enumeration Values

Numeric Value	Name	Description
0	Positive X	Positive X
1	Negative X	Negative X
2	Positive Y	Positive Y
3	Negative Y	Negative Y
4	Positive Z	Positive Z
5	Negative Z	Negative Z
6	Not Used	Not Used
7	Positive X/Y 45 degree	Positive X/Y 45 degree

Table 181: Set Wheel Configuration Message Format

Set ID/Get ID	22/137		Parameters Length (bytes)	4
Description	Set wheel configuration parameters			
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	8	RW1 Configuration	ENUM	RW1 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	8	8	RW2 Configuration	ENUM	RW2 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values

	16	8	RW3 Configuration	ENUM	RW3 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	24	8	RW4 Configuration	ENUM	RW4 or Momentum wheel Configuration. Possible values are in Table 180: AxisSelect Enumeration Values

Table 182: Set Rate Gyro Configuration Message Format

Set ID/Get ID	23/138		**Parameters Length (bytes)**		10
Description	Set rate gyro configuration parameters				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Gyro1 Configuration	ENUM	Gyro1 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	8	8	Gyro2 Configuration	ENUM	Gyro2 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	16	8	Gyro3 Configuration	ENUM	Gyro3 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	24	16	X-Rate Sensor Offset	INT	X-Rate Sensor Offset. Raw parameter value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	40	16	Y-Rate Sensor Offset	INT	Y-Rate Sensor Offset. Raw parameter value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	56	16	Z-Rate Sensor Offset	INT	Z-Rate Sensor Offset. Raw parameter value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	72	8	RateSensorMult	UINT	Multiplier of rate sensor measurement

Table 183: CSS Configuration Message Format

Set ID/Get ID	24/139		Parameters Length (bytes)		21
Description	Photodiode pointing directions and scale factors				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	CSS1 Configuration	ENUM	CSS1 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	8	8	CSS2 Configuration	ENUM	CSS2 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	16	8	CSS3 Configuration	ENUM	CSS3 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	24	8	CSS4 Configuration	ENUM	CSS4 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	32	8	CSS5 Configuration	ENUM	CSS5 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	40	8	CSS6 Configuration	ENUM	CSS6 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	48	8	CSS7 Configuration	ENUM	CSS7 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	56	8	CSS8 Configuration	ENUM	CSS8 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	64	8	CSS9 Configuration	ENUM	CSS9 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	72	8	CSS10 Configuration	ENUM	CSS10 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	80	8	CSS1 Relative Scale	UINT	CSS1 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	88	8	CSS2 Relative Scale	UINT	CSS2 Relative Scaling Factor. Raw parameter value is obtained

					using the formula: (formatted value) = RAWVAL*0.01
	96	8	CSS3 Relative Scale	UINT	CSS3 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	104	8	CSS4 Relative Scale	UINT	CSS4 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	112	8	CSS5 Relative Scale	UINT	CSS5 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	120	8	CSS6 Relative Scale	UINT	CSS6 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	128	8	CSS7 Relative Scale	UINT	CSS7 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	136	8	CSS8 Relative Scale	UINT	CSS8 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	144	8	CSS9 Relative Scale	UINT	CSS9 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	152	8	CSS10 Relative Scale	UINT	CSS10 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	160	8	CSS Threshold	UINT	CSS Threshold

Table 184: ADCS Power Control Message Format

Set ID/Get ID	11/197	**Parameters Length (bytes)**	3
Description	Control power to selected components		

Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	2	CubeControl Signal Power Selection	ENUM	Control power to electronics of CubeControl Signal PIC. Possible values are in Table 185: PowerSelect Enumeration Values
	2	2	CubeControl Motor Power Selection	ENUM	Control power to electronics of CubeControl Motor PIC. Possible values are in Table 185: PowerSelect Enumeration Values
	4	2	CubeSense1 Power Selection	ENUM	Control power to the CubeSense1. Possible values are in Table 185: PowerSelect Enumeration Values
	6	2	CubeSense2 Power Selection	ENUM	Control power to the CubeSense2. Possible values are in Table 185: PowerSelect Enumeration Values
	8	2	CubeStarPower Power Selection	ENUM	Control power to the CubeStar. Possible values are in Table 185: PowerSelect Enumeration Values
	10	2	CubeWheel1Power Power Selection	ENUM	Control power to the CubeWheel1. Possible values are in Table 185: PowerSelect Enumeration Values
	12	2	CubeWheel2Power Power Selection	ENUM	Control power to the CubeWheel2. Possible values are in Table 185: PowerSelect Enumeration Values
	14	2	CubeWheel3Power Power Selection	ENUM	Control power to the CubeWheel3. Possible values are in Table 185: PowerSelect Enumeration Values
	16	2	Motor Power	ENUM	Control power to Motor electronics. Possible values are in Table 185: PowerSelect Enumeration Values
	18	2	GPS Power	ENUM	Control power to GPS LNA. Possible values are in Table 185: PowerSelect Enumeration Values

Table 185: PowerSelect Enumeration Values

Numeric Value	Name	Description
0	Off	Off
1	On	On
2	Power state kept the same	Power state kept the same

Table 186: Commanded Attitude Angles Message Format

Set ID/Get ID	15/199		Parameters Length (bytes)		6
Description	Commanded attitude angles				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Commanded Roll Angle	INT	Commanded roll angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	16	16	Commanded Pitch Angle	INT	Commanded pitch angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	32	16	Commanded Yaw Angle	INT	Commanded yaw angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01

Table 187: Tracking Controller Target Reference Message Format

Set ID/Get ID	55/200		Parameters Length (bytes)		12
Description	Target reference for tracking control mode				
Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	32	Geocentric longitude of target	FLOAT	Geocentric longitude of target. (Unit of measure is [deg])
	32	32	Geocentric latitude of target	FLOAT	Geocentric latitude of target. (Unit of measure is [deg])
	64	32	Geocentric altitude of target	FLOAT	Geocentric altitude of target. (Unit of measure is [meter])

Table 188: Set Star Tracker Configuration Message Format

Set ID/Get ID	37/202		**Parameters Length (bytes)**	53
Description	Set configurations of CubeStar			
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	StarTracker Mounting Transform Alpha Angle	INT	StarTracker Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	16	16	StarTracker Mounting Transform Beta Angle	INT	StarTracker Mounting Transform Beta Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	32	16	StarTracker Mounting Transform Gamma Angle	INT	StarTracker Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	48	16	StarTracker exposure time	UINT	exposure time register value
	64	16	StarTracker analog gain	UINT	analog gain register value
	80	8	StarTracker detection threshold	UINT	StarTracker detection threshold
	88	8	StarTracker star threshold	UINT	StarTracker star threshold
	96	8	Maximum Star Matched	UINT	Maximum of stars that the star tracker will match
	104	16	Detection Timeout duration	UINT	Time allowed for detection
	120	8	Maximum Star Pixel	UINT	Maximum pixels in a star
	128	8	Minimum Star Pixel	UINT	Minimum pixels in a star
	136	8	Star Tracker Error Margin	UINT	% Error margin of the star identification. Raw parameter value is obtained using the formula: (formatted value) [%] = RAWVAL*0.01
	144	16	Star Tracker Delay Time	UINT	Delay Time. (Unit of measure is [milliseconds])

	160	32	Star Tracker Centroid X	FLOAT	Pixel centroid X
	192	32	Star Tracker Centroid Y	FLOAT	Pixel centroid Y
	224	32	Star Tracker Focal Length	FLOAT	Star Tracker Focal Length. (Unit of measure is [mm])
	256	32	K1 radial distortion coefficient	FLOAT	First radial distortion coefficient. (Unit of measure is [gain])
	288	32	K2 radial distortion coefficient	FLOAT	Second radial distortion coefficient. (Unit of measure is [gain])
	320	32	P1 tangential distortion coefficient	FLOAT	First tangential distortion coefficient. (Unit of measure is [gain])
	352	32	P2 tangential distortion coefficient	FLOAT	Second tangential distortion coefficients. (Unit of measure is [gain])
	384	8	Star tracking window width	UINT	Window width
	392	8	Star Tracking Margin	UINT	Tracking Margin. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL/100.0
	400	8	Star Validation Margin	UINT	Validation Margin. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL/100.0
	408	1	Star Tracking Module Enable	BOOL	Module Enable
	409	1	Star Tracking Location Prediction Enable	BOOL	LocationPredictionEnable
	416	8	Star Tracking Search Width	UINT	Search Width. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL/5.0

Table 189: CubeSense Configuration Message Format

Set ID/Get ID	25/203	**Parameters Length (bytes)**	112
Description	CubeSense configuration parameters		

Parameters	Offset (bits)	Length (bits)	Name	Data Type	Description
	0	16	Cam1 Sensor Mounting Transform Alpha Angle	INT	Cam1 Sensor Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	16	16	Cam1 Sensor Mounting Transform Beta Angle	INT	Cam1 Sensor Mounting Transform Beta Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	32	16	Cam1 Sensor Mounting Transform Gamma Angle	INT	Cam1 Sensor Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	48	8	Cam1 detection threshold	UINT	
	56	1	Cam1 sensor auto adjust mode	BOOL	0 = disabled and 1 = enabled
	64	16	Cam1 sensor exposure time	UINT	exposure time register value
	80	16	Cam1 Boresight X	UINT	X Pixel location of Cam1 boresight. Raw parameter value is obtained using the formula: (formatted value) [pixels] = RAWVAL*0.01
	96	16	Cam1 Boresight Y	UINT	Y Pixel location of Cam1 boresight. Raw parameter value is obtained using the formula: (formatted value) [pixels] = RAWVAL*0.01
	112	16	Cam2 Sensor Mounting Transform Alpha Angle	INT	Cam2 Sensor Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	128	16	Cam2 Sensor Mounting Transform Beta Angle	INT	Cam2 Sensor Mounting Transform Beta Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01

	144	16	Cam2 Sensor Mounting Transform Gamma Angle	INT	Cam2 Sensor Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	160	8	Cam2 detection threshold	UINT	Cam2 detection threshold
	168	1	Cam2 sensor auto adjust mode	BOOL	0 = disabled and 1 = enabled
	176	16	Cam2 sensor exposure time	UINT	exposure time register value
	192	16	Cam2 Boresight X	UINT	X Pixel location of Cam2 boresight. Raw parameter value is obtained using the formula: (formatted value) [pixels] = RAWVAL*0.01
	208	16	Cam2 Boresight Y	UINT	Y Pixel location of Cam2 boresight. Raw parameter value is obtained using the formula: (formatted value) [pixels] = RAWVAL*0.01
	224	8	Nadir Max Deviation Percentage	UINT	Percentage of measured angular radius as edge's maximum allowable deviation
	232	8	Nadir Max Bad Edges	UINT	Maximum amount of edges allowed outside maximum deviation (>50 to disable)
	240	8	Nadir Max Radius	UINT	Maximum Radius. (Unit of measure is [°])
	248	8	Nadir Min Radius	UINT	Minimum Radius. (Unit of measure is [°])
	256	16	Cam 1 Minimum X of area 1	UINT	Cam 1 Minimum X of Area 1
	272	16	Cam 1 Maximum X of area 1	UINT	Cam 1 Maximum X of Area 1
	288	16	Cam 1 Minimum Y of area 1	UINT	Cam 1 Minimum Y of Area 1
	304	16	Cam 1 Maximum Y of area 1	UINT	Cam 1 Maximum Y of Area 1
	320	16	Cam 1 Minimum X of area 2	UINT	Cam 1 Minimum X of Area 2

	336	16	Cam 1 Maximum X of area 2	UINT	Cam 1 Maximum X of Area 2
	352	16	Cam 1 Minimum Y of area 2	UINT	Cam 1 Minimum Y of Area 2
	368	16	Cam 1 Maximum Y of area 2	UINT	Cam 1 Maximum Y of Area 2
	384	16	Cam 1 Minimum X of area 3	UINT	Cam 1 Minimum X of Area 3
	400	16	Cam 1 Maximum X of area 3	UINT	Cam 1 Maximum X of Area 3
	416	16	Cam 1 Minimum Y of area 3	UINT	Cam 1 Minimum Y of Area 3
	432	16	Cam 1 Maximum Y of area 3	UINT	Cam 1 Maximum Y of Area 3
	448	16	Cam 1 Minimum X of area 4	UINT	Cam 1 Minimum X of Area 4
	464	16	Cam 1 Maximum X of area 4	UINT	Cam 1 Maximum X of Area 4
	480	16	Cam 1 Minimum Y of area 4	UINT	Cam 1 Minimum Y of Area 4
	496	16	Cam 1 Maximum Y of area 4	UINT	Cam 1 Maximum Y of Area 4
	512	16	Cam 1 Minimum X of area 5	UINT	Cam 1 Minimum X of Area 5
	528	16	Cam 1 Maximum X of area 5	UINT	Cam 1 Maximum X of Area 5
	544	16	Cam 1 Minimum Y of area 5	UINT	Cam 1 Minimum Y of Area 5
	560	16	Cam 1 Maximum Y of area 5	UINT	Cam 1 Maximum Y of Area 5
	576	16	Cam 2 Minimum X of area 1	UINT	Cam 2 Minimum X of Area 1
	592	16	Cam 2 Maximum X of area 1	UINT	Cam 2 Maximum X of Area 1
	608	16	Cam 2 Minimum Y of area 1	UINT	Cam 2 Minimum Y of Area 1
	624	16	Cam 2 Maximum Y of area 1	UINT	Cam 2 Maximum Y of Area 1
	640	16	Cam 2 Minimum X of area 2	UINT	Cam 2 Minimum X of Area 2
	656	16	Cam 2 Maximum X of area 2	UINT	Cam 2 Maximum X of Area 2

	672	16	Cam 2 Minimum Y of area 2	UINT	Cam 2 Minimum Y of Area 2
	688	16	Cam 2 Maximum Y of area 2	UINT	Cam 2 Maximum Y of Area 2
	704	16	Cam 2 Minimum X of area 3	UINT	Cam 2 Minimum X of Area 3
	720	16	Cam 2 Maximum X of area 3	UINT	Cam 2 Maximum X of Area 3
	736	16	Cam 2 Minimum Y of area 3	UINT	Cam 2 Minimum Y of Area 3
	752	16	Cam 2 Maximum Y of area 3	UINT	Cam 2 Maximum Y of Area 3
	768	16	Cam 2 Minimum X of area 4	UINT	Cam 2 Minimum X of Area 4
	784	16	Cam 2 Maximum X of area 4	UINT	Cam 2 Maximum X of Area 4
	800	16	Cam 2 Minimum Y of area 4	UINT	Cam 2 Minimum Y of Area 4
	816	16	Cam 2 Maximum Y of area 4	UINT	Cam 2 Maximum Y of Area 4
	832	16	Cam 2 Minimum X of area 5	UINT	Cam 2 Minimum X of Area 5
	848	16	Cam 2 Maximum X of area 5	UINT	Cam 2 Maximum X of Area 5
	864	16	Cam 2 Minimum Y of area 5	UINT	Cam 2 Minimum Y of Area 5
	880	16	Cam 2 Maximum Y of area 5	UINT	Cam 2 Maximum Y of Area 5

Table 190: Magnetometer Configuration Message Format

Set ID/Get ID	26/204		**Parameters Length (bytes)**		30
Description	Magnetometer configuration parameters				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Magnetometer Mounting Transform Alpha Angle	INT	Magnetometer Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01

	16	16	Magnetometer Mounting Transform Beta Angle	INT	Magnetometer Mounting Transform Beta Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	32	16	Magnetometer Mounting Transform Gamma Angle	INT	Magnetometer Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	48	16	Magnetometer Channel 1 Offset	INT	Magnetometer Channel 1 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	64	16	Magnetometer Channel 2 Offset	INT	Magnetometer Channel 2 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	80	16	Magnetometer Channel 3 Offset	INT	Magnetometer Channel 3 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	96	16	Magnetometer Sensitivity Matrix S11	INT	Magnetometer Sensitivity Matrix S11. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	112	16	Magnetometer Sensitivity Matrix S22	INT	Magnetometer Sensitivity Matrix S22. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	128	16	Magnetometer Sensitivity Matrix S33	INT	Magnetometer Sensitivity Matrix S33. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	144	16	Magnetometer Sensitivity Matrix S12	INT	Magnetometer Sensitivity Matrix S12. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001

	160	16	Magnetometer Sensitivity Matrix S13	INT	Magnetometer Sensitivity Matrix S13. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	176	16	Magnetometer Sensitivity Matrix S21	INT	Magnetometer Sensitivity Matrix S21. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	192	16	Magnetometer Sensitivity Matrix S23	INT	Magnetometer Sensitivity Matrix S23. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	208	16	Magnetometer Sensitivity Matrix S31	INT	Magnetometer Sensitivity Matrix S31. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	224	16	Magnetometer Sensitivity Matrix S32	INT	Magnetometer Sensitivity Matrix S32. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001

Table 191: Redundant Magnetometer Configuration Message Format

Set ID/Get ID	36/205		**Parameters Length (bytes)**		30
Description	Redundant magnetometer configuration parameters				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Redundant Magnetometer Mounting Transform Alpha Angle	INT	Redundant Magnetometer Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	16	16	Redundant Magnetometer Mounting	INT	Redundant Magnetometer Mounting Transform Beta Angle. Raw parameter value is obtained

			Transform Beta Angle		using the formula: (formatted value) [deg] = RAWVAL*0.01
	32	16	Redundant Magnetometer Mounting Transform Gamma Angle	INT	Redundant Magnetometer Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	48	16	Redundant Magnetometer Channel 1 Offset	INT	Redundant Magnetometer Channel 1 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	64	16	Redundant Magnetometer Channel 2 Offset	INT	Redundant Magnetometer Channel 2 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	80	16	Redundant Magnetometer Channel 3 Offset	INT	Redundant Magnetometer Channel 3 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	96	16	Redundant Magnetometer Sensitivity Matrix S11	INT	Redundant Magnetometer Sensitivity Matrix S11. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	112	16	Redundant Magnetometer Sensitivity Matrix S22	INT	Redundant Magnetometer Sensitivity Matrix S22. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	128	16	Redundant Magnetometer Sensitivity Matrix S33	INT	Redundant Magnetometer Sensitivity Matrix S33. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	144	16	Redundant Magnetometer Sensitivity Matrix S12	INT	Redundant Magnetometer Sensitivity Matrix S12. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001

	160	16	Redundant Magnetometer Sensitivity Matrix S13	INT	Redundant Magnetometer Sensitivity Matrix S13. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	176	16	Redundant Magnetometer Sensitivity Matrix S21	INT	Redundant Magnetometer Sensitivity Matrix S21. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	192	16	Redundant Magnetometer Sensitivity Matrix S23	INT	Redundant Magnetometer Sensitivity Matrix S23. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	208	16	Redundant Magnetometer Sensitivity Matrix S31	INT	Redundant Magnetometer Sensitivity Matrix S31. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	224	16	Redundant Magnetometer Sensitivity Matrix S32	INT	Redundant Magnetometer Sensitivity Matrix S32. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001

Table 192: ADCS Configuration Message Format

Set ID/Get ID	20/206		**Parameters Length (bytes)**		504
Description	Current configuration				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	8	Magnetorquer 1 Configuration	ENUM	Magnetorquer 1 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	8	8	Magnetorquer 2 Configuration	ENUM	Magnetorquer 2 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	16	8	Magnetorquer 3 Configuration	ENUM	Magnetorquer 3 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values

	24	8	RW1 Configuration	ENUM	RW1 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	32	8	RW2 Configuration	ENUM	RW2 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	40	8	RW3 Configuration	ENUM	RW3 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	48	8	RW4 Configuration	ENUM	RW4 or Momentum wheel Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	56	8	Gyro1 Configuration	ENUM	Gyro1 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	64	8	Gyro2 Configuration	ENUM	Gyro2 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	72	8	Gyro3 Configuration	ENUM	Gyro3 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	80	16	X-Rate Sensor Offset	INT	X-Rate Sensor Offset. Raw parameter value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	96	16	Y-Rate Sensor Offset	INT	Y-Rate Sensor Offset. Raw parameter value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	112	16	Z-Rate Sensor Offset	INT	Z-Rate Sensor Offset. Raw parameter value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	128	8	RateSensorMult	UINT	Multiplier of rate sensor measurement
	136	8	CSS1 Configuration	ENUM	CSS1 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	144	8	CSS2 Configuration	ENUM	CSS2 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values

	152	8	CSS3 Configuration	ENUM	CSS3 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	160	8	CSS4 Configuration	ENUM	CSS4 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	168	8	CSS5 Configuration	ENUM	CSS5 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	176	8	CSS6 Configuration	ENUM	CSS6 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	184	8	CSS7 Configuration	ENUM	CSS7 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	192	8	CSS8 Configuration	ENUM	CSS8 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	200	8	CSS9 Configuration	ENUM	CSS9 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	208	8	CSS10 Configuration	ENUM	CSS10 Configuration. Possible values are in Table 180: AxisSelect Enumeration Values
	216	8	CSS1 Relative Scale	UINT	CSS1 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	224	8	CSS2 Relative Scale	UINT	CSS2 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	232	8	CSS3 Relative Scale	UINT	CSS3 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	240	8	CSS4 Relative Scale	UINT	CSS4 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	248	8	CSS5 Relative Scale	UINT	CSS5 Relative Scaling Factor. Raw parameter value is obtained

					using the formula: (formatted value) = RAWVAL*0.01
	256	8	CSS6 Relative Scale	UINT	CSS6 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	264	8	CSS7 Relative Scale	UINT	CSS7 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	272	8	CSS8 Relative Scale	UINT	CSS8 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	280	8	CSS9 Relative Scale	UINT	CSS9 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	288	8	CSS10 Relative Scale	UINT	CSS10 Relative Scaling Factor. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	296	8	CSS Threshold	UINT	CSS Threshold
	304	16	Cam1 Sensor Mounting Transform Alpha Angle	INT	Cam1 Sensor Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	320	16	Cam1 Sensor Mounting Transform Beta Angle	INT	Cam1 Sensor Mounting Transform Beta Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	336	16	Cam1 Sensor Mounting Transform Gamma Angle	INT	Cam1 Sensor Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	352	8	Cam1 detection threshold	UINT	Cam1 detection threshold
	360	1	Cam1 sensor auto adjust mode	BOOL	0 = disabled and 1 = enabled

	368	16	Cam1 sensor exposure time	UINT	exposure time register value
	384	16	Cam1 Boresight X	UINT	X Pixel location of Cam1 boresight. Raw parameter value is obtained using the formula: (formatted value) [pixels] = RAWVAL*0.01
	400	16	Cam1 Boresight Y	UINT	Y Pixel location of Cam1 boresight. Raw parameter value is obtained using the formula: (formatted value) [pixels] = RAWVAL*0.01
	416	16	Cam2 Sensor Mounting Transform Alpha Angle	INT	Cam2 Sensor Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	432	16	Cam2 Sensor Mounting Transform Beta Angle	INT	Cam2 Sensor Mounting Transform Beta Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	448	16	Cam2 Sensor Mounting Transform Gamma Angle	INT	Cam2 Sensor Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	464	8	Cam2 detection threshold	UINT	Cam2 detection threshold
	472	1	Cam2 sensor auto adjust mode	BOOL	0 = disabled and 1 = enabled
	480	16	Cam2 sensor exposure time	UINT	exposure time register value
	496	16	Cam2 Boresight X	UINT	X Pixel location of Cam2 boresight. Raw parameter value is obtained using the formula: (formatted value) [pixels] = RAWVAL*0.01
	512	16	Cam2 Boresight Y	UINT	Y Pixel location of Cam2 boresight. Raw parameter value is obtained using the formula: (formatted value) [pixels] = RAWVAL*0.01

	528	8	Nadir Max Deviation Percentage	UINT	Percentage of measured angular radius as edge's maximum allowable deviation
	536	8	Nadir Max Bad Edges	UINT	Maximum amount of edges allowed outside maximum deviation (>50 to disable)
	544	8	Nadir Max Radius	UINT	Maximum Radius. (Unit of measure is [°])
	552	8	Nadir Min Radius	UINT	Minimum Radius. (Unit of measure is [°])
	560	16	Cam 1 Minimum X of area 1	UINT	Cam 1 Minimum X of Area 1
	576	16	Cam 1 Maximum X of area 1	UINT	Cam 1 Maximum X of Area 1
	592	16	Cam 1 Minimum Y of area 1	UINT	Cam 1 Minimum Y of Area 1
	608	16	Cam 1 Maximum Y of area 1	UINT	Cam 1 Maximum Y of Area 1
	624	16	Cam 1 Minimum X of area 2	UINT	Cam 1 Minimum X of Area 2
	640	16	Cam 1 Maximum X of area 2	UINT	Cam 1 Maximum X of Area 2
	656	16	Cam 1 Minimum Y of area 2	UINT	Cam 1 Minimum Y of Area 2
	672	16	Cam 1 Maximum Y of area 2	UINT	Cam 1 Maximum Y of Area 2
	688	16	Cam 1 Minimum X of area 3	UINT	Cam 1 Minimum X of Area 3
	704	16	Cam 1 Maximum X of area 3	UINT	Cam 1 Maximum X of Area 3
	720	16	Cam 1 Minimum Y of area 3	UINT	Cam 1 Minimum Y of Area 3
	736	16	Cam 1 Maximum Y of area 3	UINT	Cam 1 Maximum Y of Area 3
	752	16	Cam 1 Minimum X of area 4	UINT	Cam 1 Minimum X of Area 4
	768	16	Cam 1 Maximum X of area 4	UINT	Cam 1 Maximum X of Area 4
	784	16	Cam 1 Minimum Y of area 4	UINT	Cam 1 Minimum Y of Area 4
	800	16	Cam 1 Maximum Y of area 4	UINT	Cam 1 Maximum Y of Area 4

	816	16	Cam 1 Minimum X of area 5	UINT	Cam 1 Minimum X of Area 5
	832	16	Cam 1 Maximum X of area 5	UINT	Cam 1 Maximum X of Area 5
	848	16	Cam 1 Minimum Y of area 5	UINT	Cam 1 Minimum Y of Area 5
	864	16	Cam 1 Maximum Y of area 5	UINT	Cam 1 Maximum Y of Area 5
	880	16	Cam 2 Minimum X of area 1	UINT	Cam 2 Minimum X of Area 1
	896	16	Cam 2 Maximum X of area 1	UINT	Cam 2 Maximum X of Area 1
	912	16	Cam 2 Minimum Y of area 1	UINT	Cam 2 Minimum Y of Area 1
	928	16	Cam 2 Maximum Y of area 1	UINT	Cam 2 Maximum Y of Area 1
	944	16	Cam 2 Minimum X of area 2	UINT	Cam 2 Minimum X of Area 2
	960	16	Cam 2 Maximum X of area 2	UINT	Cam 2 Maximum X of Area 2
	976	16	Cam 2 Minimum Y of area 2	UINT	Cam 2 Minimum Y of Area 2
	992	16	Cam 2 Maximum Y of area 2	UINT	Cam 2 Maximum Y of Area 2
	1008	16	Cam 2 Minimum X of area 3	UINT	Cam 2 Minimum X of Area 3
	1024	16	Cam 2 Maximum X of area 3	UINT	Cam 2 Maximum X of Area 3
	1040	16	Cam 2 Minimum Y of area 3	UINT	Cam 2 Minimum Y of Area 3
	1056	16	Cam 2 Maximum Y of area 3	UINT	Cam 2 Maximum Y of Area 3
	1072	16	Cam 2 Minimum X of area 4	UINT	Cam 2 Minimum X of Area 4
	1088	16	Cam 2 Maximum X of area 4	UINT	Cam 2 Maximum X of Area 4
	1104	16	Cam 2 Minimum Y of area 4	UINT	Cam 2 Minimum Y of Area 4
	1120	16	Cam 2 Maximum Y of area 4	UINT	Cam 2 Maximum Y of Area 4
	1136	16	Cam 2 Minimum X of area 5	UINT	Cam 2 Minimum X of Area 5

	1152	16	Cam 2 Maximum X of area 5	UINT	Cam 2 Maximum X of Area 5
	1168	16	Cam 2 Minimum Y of area 5	UINT	Cam 2 Minimum Y of Area 5
	1184	16	Cam 2 Maximum Y of area 5	UINT	Cam 2 Maximum Y of Area 5
	1200	16	Magnetometer Mounting Transform Alpha Angle	INT	Magnetometer Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	1216	16	Magnetometer Mounting Transform Beta Angle	INT	Magnetometer Mounting Transform Beta Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	1232	16	Magnetometer Mounting Transform Gamma Angle	INT	Magnetometer Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	1248	16	Magnetometer Channel 1 Offset	INT	Magnetometer Channel 1 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1264	16	Magnetometer Channel 2 Offset	INT	Magnetometer Channel 2 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1280	16	Magnetometer Channel 3 Offset	INT	Magnetometer Channel 3 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1296	16	Magnetometer Sensitivity Matrix S11	INT	Magnetometer Sensitivity Matrix S11. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1312	16	Magnetometer Sensitivity Matrix S22	INT	Magnetometer Sensitivity Matrix S22. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001

	1328	16	Magnetometer Sensitivity Matrix S33	INT	Magnetometer Sensitivity Matrix S33. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1344	16	Magnetometer Sensitivity Matrix S12	INT	Magnetometer Sensitivity Matrix S12. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1360	16	Magnetometer Sensitivity Matrix S13	INT	Magnetometer Sensitivity Matrix S13. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1376	16	Magnetometer Sensitivity Matrix S21	INT	Magnetometer Sensitivity Matrix S21. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1392	16	Magnetometer Sensitivity Matrix S23	INT	Magnetometer Sensitivity Matrix S23. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1408	16	Magnetometer Sensitivity Matrix S31	INT	Magnetometer Sensitivity Matrix S31. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1424	16	Magnetometer Sensitivity Matrix S32	INT	Magnetometer Sensitivity Matrix S32. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1440	16	Redundant Magnetometer Mounting Transform Alpha Angle	INT	Redundant Magnetometer Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	1456	16	Redundant Magnetometer	INT	Redundant Magnetometer Mounting Transform Beta Angle.

			Mounting Transform Beta Angle		Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	1472	16	Redundant Magnetometer Mounting Transform Gamma Angle	INT	Redundant Magnetometer Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	1488	16	Redundant Magnetometer Channel 1 Offset	INT	Redundant Magnetometer Channel 1 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1504	16	Redundant Magnetometer Channel 2 Offset	INT	Redundant Magnetometer Channel 2 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1520	16	Redundant Magnetometer Channel 3 Offset	INT	Redundant Magnetometer Channel 3 Offset. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1536	16	Redundant Magnetometer Sensitivity Matrix S11	INT	Redundant Magnetometer Sensitivity Matrix S11. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1552	16	Redundant Magnetometer Sensitivity Matrix S22	INT	Redundant Magnetometer Sensitivity Matrix S22. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1568	16	Redundant Magnetometer Sensitivity Matrix S33	INT	Redundant Magnetometer Sensitivity Matrix S33. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1584	16	Redundant Magnetometer Sensitivity Matrix S12	INT	Redundant Magnetometer Sensitivity Matrix S12. Raw parameter value is obtained

					using the formula: (formatted value) = RAWVAL*0.001
	1600	16	Redundant Magnetometer Sensitivity Matrix S13	INT	Redundant Magnetometer Sensitivity Matrix S13. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1616	16	Redundant Magnetometer Sensitivity Matrix S21	INT	Redundant Magnetometer Sensitivity Matrix S21. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1632	16	Redundant Magnetometer Sensitivity Matrix S23	INT	Redundant Magnetometer Sensitivity Matrix S23. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1648	16	Redundant Magnetometer Sensitivity Matrix S31	INT	Redundant Magnetometer Sensitivity Matrix S31. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1664	16	Redundant Magnetometer Sensitivity Matrix S32	INT	Redundant Magnetometer Sensitivity Matrix S32. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	1680	16	StarTracker Mounting Transform Alpha Angle	INT	StarTracker Mounting Transform Alpha Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	1696	16	StarTracker Mounting Transform Beta Angle	INT	StarTracker Mounting Transform Beta Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01
	1712	16	StarTracker Mounting Transform Gamma Angle	INT	StarTracker Mounting Transform Gamma Angle. Raw parameter value is obtained using the formula: (formatted value) [deg] = RAWVAL*0.01

	1728	16	StarTracker exposure time	UINT	exposure time register value
	1744	16	StarTracker analog gain	UINT	analog gain register value
	1760	8	StarTracker detection threshold	UINT	StarTracker detection threshold
	1768	8	StarTracker star threshold	UINT	StarTracker star threshold
	1776	8	Maximum Star Matched	UINT	Maximum of stars that the star tracker will match
	1784	16	Detection Timeout duration	UINT	Time allowed for detection
	1800	8	Maximum Star Pixel	UINT	Maximum pixels in a star
	1808	8	Minimum Star Pixel	UINT	Minimum pixels in a star
	1816	8	Star Tracker Error Margin	UINT	% Error margin of the star identification. Raw parameter value is obtained using the formula: (formatted value) [%] = RAWVAL*0.01
	1824	16	Star Tracker Delay Time	UINT	Delay Time. (Unit of measure is [milliseconds])
	1840	32	Star Tracker Centroid X	FLOAT	Pixel centroid X
	1872	32	Star Tracker Centroid Y	FLOAT	Pixel centroid Y
	1904	32	Star Tracker Focal Length	FLOAT	Star Tracker Focal Length. (Unit of measure is [mm])
	1936	32	K1 radial distortion coefficient	FLOAT	First radial distortion coefficient. (Unit of measure is [gain])
	1968	32	K2 radial distortion coefficient	FLOAT	Second radial distortion coefficient. (Unit of measure is [gain])
	2000	32	P1 tangential distortion coefficient	FLOAT	First tangential distortion coefficient. (Unit of measure is [gain])
	2032	32	P2 tangential distortion coefficient	FLOAT	Second tangential distortion coefficients. (Unit of measure is [gain])
	2064	8	Star tracking window width	UINT	Window width
	2072	8	Star Tracking Margin	UINT	Tracking Margin. Raw parameter value is obtained using the

					formula: (formatted value) = RAWVAL/100.0
	2080	8	Star Validation Margin	UINT	Validation Margin. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL/100.0
	2088	1	Star Tracking Module Enable	BOOL	Module Enable
	2089	1	Star Tracking Location Prediction Enable	BOOL	LocationPredictionEnable
	2096	8	Star Tracking Search Width	UINT	Search Width. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL/5.0
	2104	32	Detumbling Spin Gain	FLOAT	Detumbling Spin Gain (Ks)
	2136	32	Detumbling Damping Gain	FLOAT	Detumbling Damping Gain (Kd)
	2168	16	Reference spin rate	INT	Reference spin rate (wy-ref). Must always be smaller than 0 for Y-spin. Raw parameter value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	2184	32	Fast Bdot Detumbling Gain	FLOAT	Fast Bdot Detumbling Gain (Kdf)
	2216	32	Y-Momentum Control Gain	FLOAT	Y-Momentum Control Gain (Kh)
	2248	32	Y-momentum Nutation Damping Gain	FLOAT	Y-momentum Nutation Damping Gain (Kn)
	2280	32	Y-momentum Proportional Gain	FLOAT	Y-momentum Proportional Gain (Kp1)
	2312	32	Y-momentum Derivative Gain	FLOAT	Y-momentum Derivative Gain (Kd1)
	2344	32	Reference Wheel Momentum	FLOAT	Reference Wheel Momentum (H-ref). Must always be smaller than 0. (Unit of measure is [Nms])
	2376	32	Rwheel Proportional Gain	FLOAT	Rwheel Proportional Gain (Kp2)

	2408	32	Rwheel Derivative Gain	FLOAT	Rwheel Derivative Gain (Kd2)
	2440	32	Y-Wheel Bias Momentum	FLOAT	Y-Wheel Bias Momentum (H-bias). (Unit of measure is [Nms])
	2472	7	Sun-pointing Facet	ENUM	Satellite body axis that will align with sun vector. Possible values are in Table 180: AxisSelect Enumeration Values
	2479	1	Automatic Control Transition due to Wheel Errors	BOOL	Enable/disable automatic transition from wheel control modes to Y-Thomson mode in case of wheel error
	2480	32	Tracking Proportional Gain	FLOAT	Tracking Proportional Gain (Kp3)
	2512	32	Tracking Derivative Gain	FLOAT	Tracking Derivative Gain (Kd3)
	2544	32	Tracking Integral Gain	FLOAT	Tracking Integral Gain (Ki3)
	2576	8	Target-tracking Facet	ENUM	Satellite body axis that will point to target. Possible values are in Table 180: AxisSelect Enumeration Values
	2584	32	Moment Of Inertia – Ixx	FLOAT	Moment Of Inertia – Ixx. (Unit of measure is [kg.m^2])
	2616	32	Moment Of Inertia – Iyy	FLOAT	Moment Of Inertia – Iyy. (Unit of measure is [kg.m^2])
	2648	32	Moment Of Inertia – Izz	FLOAT	Moment Of Inertia – Izz. (Unit of measure is [kg.m^2])
	2680	32	Product Of Inertia – Ixy	FLOAT	Product Of Inertia – Ixy. (Unit of measure is [kg.m^2])
	2712	32	Product Of Inertia – Ixz	FLOAT	Product Of Inertia – Ixz. (Unit of measure is [kg.m^2])
	2744	32	Product Of Inertia – Iyz	FLOAT	Product Of Inertia – Iyz. (Unit of measure is [kg.m^2])
	2776	32	Magnetometer Rate Filter System Noise	FLOAT	Magnetometer Rate Filter System Noise
	2808	32	EKF System Noise	FLOAT	EKF System Noise
	2840	32	CSS Measurement Noise	FLOAT	CSS Measurement Noise

	2872	32	Sun Sensor Measurement Noise	FLOAT	Sun Sensor Measurement Noise
	2904	32	Nadir Sensor Measurement Noise	FLOAT	Nadir Sensor Measurement Noise
	2936	32	Magnetometer Measurement Noise	FLOAT	Magnetometer Measurement Noise
	2968	32	Star Tracker Measurement Noise	FLOAT	Star Tracker Measurement Noise
	3000	1	Use Sun Sensor	BOOL	Use Sun Sensor measurement in EKF
	3001	1	Use Nadir Sensor	BOOL	Use Nadir Sensor measurement in EKF
	3002	1	Use CSS	BOOL	Use CSS measurement in EKF
	3003	1	Use Star Tracker	BOOL	Use Star Tracker measurement in EKF
	3004	1	Nadir sensor terminator test	BOOL	Select to ignore Nadir sensor measurements when terminator is in FOV
	3005	1	Automatic Magnetometer Recovery	BOOL	Select whether automatic switch to redundant magnetometer should occur in case of failure
	3006	2	Magnetometer Mode	ENUM	Mode describing which magnetometer is used for estimation and control. Possible values are in Table 90: MagModeVal Enumeration Values
	3008	2	Magnetometer Selection for RAW MTM TLM	ENUM	Select which magnetometer and sampling to use for 2nd raw magnetometer tlm frame. Possible values are in Table 90: MagModeVal Enumeration Values
	3010	1	Automatic Estimation Transition due to Rate Sensor Errors	BOOL	Enable/disable automatic transition from MEMS rate estimation mode to RKF in case of rate sensor error

	3016	8	Cam1 and Cam2 Sampling Period	UINT	Cam1 and Cam2 sensor sampling period. Lower four bits are Cam1 period and upper four bits the Cam2 period. Setting period to zero for sensor will disable sampling of sensor. (Unit of measure is [s])
	3024	16	Incl. coefficient	UINT	Inclination filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	3040	16	Raan coefficient	UINT	RAAN filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	3056	16	Ecc coefficient	UINT	Eccentricity filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	3072	16	Aop coefficient	UINT	Argument of perigee filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	3088	16	Time coefficient	UINT	Time filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	3104	16	Pos coefficient	UINT	Position filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	3120	8	Maximum position error	UINT	Maximum position error for asgp4 to continue working. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.1
	3128	8	ASGP4 filter	ENUM	The type of filter asgp4 is using. Possible values are in Table 193: AsgpFilter Enumeration Values
	3136	32	xp coefficient	INT	Polar coefficient xp. Raw parameter value is obtained

					using the formula: (formatted value) = RAWVAL*0.0000001
	3168	32	yp coefficient	INT	Polar coefficient yp. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.0000001
	3200	8	GPS roll over	UINT	GPS roll over number
	3208	8	Position sd	UINT	Maximum position standard deviation for asgp4 to operate. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.1
	3216	8	Velocity sd	UINT	Maximum velocity standard deviation for asgp4 to operate. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	3224	8	Min. satellites	UINT	Minimum satellites for asgp4 to operate
	3232	8	Time gain	UINT	Time offset compensation gain. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	3240	8	Max. lag	UINT	Maximum lagged timestamp measurements to incorporate. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	3248	16	Min. samples	UINT	Minimum samples to use for asgp4 process
	3264	384	User Coded Controller Settings	ARRAY	Settings for user coded control mode(s)
	3648	384	User Coded Estimator Settings	ARRAY	Settings for user coded estimation mode(s)

Table 193: AsgpFilter Enumeration Values

Numeric Value	Name	Description
0	LPF	Use LPF
1	Average	Use averaging filter

Table 194: SGP4 Orbit Parameters Message Format

Set ID/Get ID	45/207		Parameters Length (bytes)		64
Description	SGP4 Orbit Parameters				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	64	Inclination	DOUBLE	Inclination. (Unit of measure is [deg])
	64	64	Eccentricity	DOUBLE	Eccentricity
	128	64	Right-ascension of the Ascending Node	DOUBLE	Right-ascension of the Ascending Node. (Unit of measure is [deg])
	192	64	Argument of Perigee	DOUBLE	Argument of Perigee. (Unit of measure is [deg])
	256	64	B-Star drag term	DOUBLE	B-Star drag term
	320	64	Mean Motion	DOUBLE	Mean Motion. (Unit of measure is [orbits/day])
	384	64	Mean Anomaly	DOUBLE	Mean Anomaly. (Unit of measure is [deg])
	448	64	Epoch	DOUBLE	Epoch (year.day). (Unit of measure is [year.day])

Table 195: Set Detumbling Control Parameters Message Format

Set ID/Get ID	38/208		Parameters Length (bytes)		14
Description	Set controller gains and reference values for Detumbling control mode				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	Detumbling Spin Gain	FLOAT	Detumbling Spin Gain (Ks)
	32	32	Detumbling Damping Gain	FLOAT	Detumbling Damping Gain (Kd)
	64	16	Reference spin rate	INT	Reference spin rate (wy-ref). Must always be smaller than 0 for Y-spin. Raw parameter value is obtained using the formula: (formatted value) [deg/s] = RAWVAL*0.001
	80	32	Fast Bdot Detumbling Gain	FLOAT	Fast Bdot Detumbling Gain (Kdf)

Table 196: Set Y-Wheel Control Parameters Message Format

Set ID/Get ID	39/209		Parameters Length (bytes)		20
Description	Set controller gains and reference value for Y-wheel control mode				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	Y-Momentum Control Gain	FLOAT	Y-Momentum Control Gain (Kh)
	32	32	Y-momentum Nutation Damping Gain	FLOAT	Y-momentum Nutation Damping Gain (Kn)
	64	32	Y-momentum Proportional Gain	FLOAT	Y-momentum Proportional Gain (Kp1)
	96	32	Y-momentum Derivative Gain	FLOAT	Y-momentum Derivative Gain (Kd1)
	128	32	Reference Wheel Momentum	FLOAT	Reference Wheel Momentum (H-ref). Must always be smaller than 0. (Unit of measure is [Nms])

Table 197: Set Reaction Wheel Control Parameters Message Format

Set ID/Get ID	40/217		Parameters Length (bytes)		13
Description	Set controller gains and reference value for reaction wheel control mode				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	Rwheel Proportional Gain	FLOAT	Rwheel Proportional Gain (Kp2)
	32	32	Rwheel Derivative Gain	FLOAT	Rwheel Derivative Gain (Kd2)
	64	32	Y-Wheel Bias Momentum	FLOAT	Y-Wheel Bias Momentum (H-bias). (Unit of measure is [Nms])
	96	7	Sun-pointing Facet	ENUM	Satellite body axis that will align with sun vector. Possible values are in Table 180: AxisSelect Enumeration Values
	103	1	Automatic Control Transition due to Wheel Errors	BOOL	Enable/disable automatic transition from wheel control modes to Y-Thomson mode in case of wheel error

Table 198: Set Tracking Controller Gain Parameters Message Format

Set ID/Get ID	54/221		**Parameters Length (bytes)**		13
Description	Set controller gains for tracking control mode				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	Tracking Proportional Gain	FLOAT	Tracking Proportional Gain (Kp3)
	32	32	Tracking Derivative Gain	FLOAT	Tracking Derivative Gain (Kd3)
	64	32	Tracking Integral Gain	FLOAT	Tracking Integral Gain (Ki3)
	96	8	Target-tracking Facet	ENUM	Satellite body axis that will point to target. Possible values are in Table 180: AxisSelect Enumeration Values

Table 199: Moment of Inertia Matrix Message Format

Set ID/Get ID	41/222		**Parameters Length (bytes)**		24
Description	Satellite moment of inertia matrix				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	Moment Of Inertia – Ixx	FLOAT	Moment Of Inertia – Ixx. (Unit of measure is [kg.m^2])
	32	32	Moment Of Inertia – Iyy	FLOAT	Moment Of Inertia – Iyy. (Unit of measure is [kg.m^2])
	64	32	Moment Of Inertia – Izz	FLOAT	Moment Of Inertia – Izz. (Unit of measure is [kg.m^2])
	96	32	Product Of Inertia – Ixy	FLOAT	Product Of Inertia – Ixy. (Unit of measure is [kg.m^2])
	128	32	Product Of Inertia – Ixz	FLOAT	Product Of Inertia – Ixz. (Unit of measure is [kg.m^2])
	160	32	Product Of Inertia – Iyz	FLOAT	Product Of Inertia – Iyz. (Unit of measure is [kg.m^2])

Table 200: Estimation Parameters Message Format

Set ID/Get ID	27/223		Parameters Length (bytes)		31
Description	Estimation noise covariance and sensor mask				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	32	Magnetometer Rate Filter System Noise	FLOAT	Magnetometer Rate Filter System Noise
	32	32	EKF System Noise	FLOAT	EKF System Noise
	64	32	CSS Measurement Noise	FLOAT	CSS Measurement Noise
	96	32	Sun Sensor Measurement Noise	FLOAT	Sun Sensor Measurement Noise
	128	32	Nadir Sensor Measurement Noise	FLOAT	Nadir Sensor Measurement Noise
	160	32	Magnetometer Measurement Noise	FLOAT	Magnetometer Measurement Noise
	192	32	Star Tracker Measurement Noise	FLOAT	Star Tracker Measurement Noise
	224	1	Use Sun Sensor	BOOL	Use Sun Sensor measurement in EKF
	225	1	Use Nadir Sensor	BOOL	Use Nadir Sensor measurement in EKF
	226	1	Use CSS	BOOL	Use CSS measurement in EKF
	227	1	Use Star Tracker	BOOL	Use Star Tracker measurement in EKF
	228	1	Nadir sensor terminator test	BOOL	Select to ignore Nadir sensor measurements when terminator is in FOV
	229	1	Automatic Magnetometer Recovery	BOOL	Select whether automatic switch to redundant magnetometer should occur in case of failure
	230	2	Magnetometer Mode	ENUM	Mode describing which magnetometer is used for estimation and control. Possible values are in Table 90:

					MagModeVal Enumeration Values
	232	2	Magnetometer Selection for RAW MTM TLM	ENUM	Select which magnetometer and sampling to use for 2nd raw magnetometer tlm frame. Possible values are in Table 90: MagModeVal Enumeration Values
	234	1	Automatic Estimation Transition due to Rate Sensor Errors	BOOL	Enable/disable automatic transition from MEMS rate estimation mode to RKF in case of rate sensor error
	240	8	Cam1 and Cam2 Sampling Period	UINT	Cam1 and Cam2 sensor sampling period. Lower four bits are Cam1 period and upper four bits the Cam2 period. Setting period to zero for sensor will disable sampling of sensor. (Unit of measure is [s])

Table 201: ADCS System Configuration Message Format

Set ID/Get ID	30/225		**Parameters Length (bytes)**		173
Description	Current hard-coded system configuration				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	4	ACP Type	ENUM	ACP Type. Possible values are in Table 202: AcpProgramType Enumeration Values
	4	4	Special Control Selection	ENUM	Special Control Selection. Possible values are in Table 203: SpecialConSelect Enumeration Values
	8	8	CubeControl Signal Version	UINT	CubeControl Signal Version
	16	8	CubeControl Motor Version	UINT	CubeControl Motor Version
	24	8	CubeSense1 Version	UINT	CubeSense1 Version
	32	8	CubeSense2 Version	UINT	CubeSense2 Version

	40	4	CubeSense1 Camera Type	ENUM	CubeSense1 Camera Type. Possible values are in Table 204: CsCamType Enumeration Values
	44	4	CubeSense2 Camera Type	ENUM	CubeSense2 Camera Type. Possible values are in Table 204: CsCamType Enumeration Values
	48	8	CubeStar Version	UINT	CubeStar Version
	56	4	GPS Type	ENUM	GPS Type. Possible values are in Table 205: GpsSelect Enumeration Values
	60	1	Redundant MTM Included	BOOL	Redundant MTM Included
	64	32	Magnetorquer-X Max Dipole	FLOAT	Magnetorquer-X Max Dipole. (Unit of measure is [A.m^2])
	96	32	Magnetorquer-Y Max Dipole	FLOAT	Magnetorquer-Y Max Dipole. (Unit of measure is [A.m^2])
	128	32	Magnetorquer-Z Max Dipole	FLOAT	Magnetorquer-Z Max Dipole. (Unit of measure is [A.m^2])
	160	32	Magnetorquer On-time Resolution	FLOAT	Magnetorquer On-time Resolution. (Unit of measure is [s])
	192	32	Magnetorquer Maximum On-time	FLOAT	Magnetorquer Maximum On-time. (Unit of measure is [s])
	224	32	RW-X Maximum Torque	FLOAT	RW-X Maximum Torque. (Unit of measure is [N.m])
	256	32	RW-Y Maximum Torque	FLOAT	RW-Y Maximum Torque. (Unit of measure is [N.m])
	288	32	RW-Z Maximum Torque	FLOAT	RW-Z Maximum Torque. (Unit of measure is [N.m])
	320	32	RW-X Maximum Momentum	FLOAT	RW-X Maximum Momentum. (Unit of measure is [Nms])
	352	32	RW-Y Maximum Momentum	FLOAT	RW-Y Maximum Momentum. (Unit of measure is [Nms])
	384	32	RW-Z Maximum Momentum	FLOAT	RW-Z Maximum Momentum. (Unit of measure is [Nms])
	416	32	RW-X Inertia	FLOAT	RW-X Inertia. (Unit of measure is [kg.m^2])
	448	32	RW-Y Inertia	FLOAT	RW-Y Inertia. (Unit of measure is [kg.m^2])
	480	32	RW-Z Inertia	FLOAT	RW-Z Inertia. (Unit of measure is [kg.m^2])

	512	32	RW Torque Increment	FLOAT	RW Torque Increment. (Unit of measure is [N.m])
	544	32	Primary MTM X-Bias d1	FLOAT	Primary MTM X-Bias d1
	576	32	Primary MTM Y-Bias d1	FLOAT	Primary MTM Y-Bias d1
	608	32	Primary MTM Z-Bias d1	FLOAT	Primary MTM Z-Bias d1
	640	32	Primary MTM X-Bias d2	FLOAT	Primary MTM X-Bias d2
	672	32	Primary MTM Y-Bias d2	FLOAT	Primary MTM Y-Bias d2
	704	32	Primary MTM Z-Bias d2	FLOAT	Primary MTM Z-Bias d2
	736	32	Primary MTM X-Sens s1	FLOAT	Primary MTM X-Sens s1
	768	32	Primary MTM Y-Sens s1	FLOAT	Primary MTM Y-Sens s1
	800	32	Primary MTM Z-Sens s1	FLOAT	Primary MTM Z-Sens s1
	832	32	Primary MTM X-Sens s2	FLOAT	Primary MTM X-Sens s2
	864	32	Primary MTM Y-Sens s2	FLOAT	Primary MTM Y-Sens s2
	896	32	Primary MTM Z-Sens s2	FLOAT	Primary MTM Z-Sens s2
	928	32	Redundant MTM X-Bias d1	FLOAT	Redundant MTM X-Bias d1
	960	32	Redundant MTM Y-Bias d1	FLOAT	Redundant MTM Y-Bias d1
	992	32	Redundant MTM Z-Bias d1	FLOAT	Redundant MTM Z-Bias d1
	1024	32	Redundant MTM X-Bias d2	FLOAT	Redundant MTM X-Bias d2
	1056	32	Redundant MTM Y-Bias d2	FLOAT	Redundant MTM Y-Bias d2
	1088	32	Redundant MTM Z-Bias d2	FLOAT	Redundant MTM Z-Bias d2
	1120	32	Redundant MTM X-Sens s1	FLOAT	Redundant MTM X-Sens s1
	1152	32	Redundant MTM Y-Sens s1	FLOAT	Redundant MTM Y-Sens s1

	1184	32	Redundant MTM Z-Sens s1	FLOAT	Redundant MTM Z-Sens s1
	1216	32	Redundant MTM X-Sens s2	FLOAT	Redundant MTM X-Sens s2
	1248	32	Redundant MTM Y-Sens s2	FLOAT	Redundant MTM Y-Sens s2
	1280	32	Redundant MTM Z-Sens s2	FLOAT	Redundant MTM Z-Sens s2
	1312	4	CC Signal Enable Port	ENUM	CC Signal Enable GPIO port. Possible values are in Table 206: GpioPort Enumeration Values
	1316	4	CC Signal Enable Pin	ENUM	CC Signal Enable GPIO port pin. Possible values are in Table 207: GpioPortPin Enumeration Values
	1320	4	CC Motor Enable Port	ENUM	CC Motor Enable GPIO port. Possible values are in Table 206: GpioPort Enumeration Values
	1324	4	CC Motor Enable Pin	ENUM	CC Motor Enable GPIO port pin. Possible values are in Table 207: GpioPortPin Enumeration Values
	1328	4	CC Common Enable Port	ENUM	CC Common Enable GPIO port. Possible values are in Table 206: GpioPort Enumeration Values
	1332	4	CC Common Enable Pin	ENUM	CC Common Enable GPIO port pin. Possible values are in Table 207: GpioPortPin Enumeration Values
	1336	4	CubeSense1 Enable Port	ENUM	CubeSense1 Enable GPIO port. Possible values are in Table 206: GpioPort Enumeration Values
	1340	4	CubeSense1 Enable Pin	ENUM	CubeSense1 Enable GPIO port pin. Possible values are in Table 207: GpioPortPin Enumeration Values
	1344	4	CubeSense2 Enable Port	ENUM	CubeSense2 Enable GPIO port. Possible values are in Table 206: GpioPort Enumeration Values
	1348	4	CubeSense2 Enable Pin	ENUM	CubeSense2 Enable GPIO port pin. Possible values are in Table 207: GpioPortPin Enumeration Values

	1352	4	CubeStar Enable Port	ENUM	CubeStar Enable GPIO port. Possible values are in Table 206: GpioPort Enumeration Values
	1356	4	CubeStar Enable Pin	ENUM	CubeStar Enable GPIO port pin. Possible values are in Table 207: GpioPortPin Enumeration Values
	1360	4	CubeWheel1 Enable Port	ENUM	CubeWheel1 Enable GPIO port. Possible values are in Table 206: GpioPort Enumeration Values
	1364	4	CubeWheel1 Enable Pin	ENUM	CubeWheel1 Enable GPIO port pin. Possible values are in Table 207: GpioPortPin Enumeration Values
	1368	4	CubeWheel2 Enable Port	ENUM	CubeWheel2 Enable GPIO port. Possible values are in Table 206: GpioPort Enumeration Values
	1372	4	CubeWheel2 Enable Pin	ENUM	CubeWheel2 Enable GPIO port pin. Possible values are in Table 207: GpioPortPin Enumeration Values
	1376	4	CubeWheel3 Enable Port	ENUM	CubeWheel3 Enable GPIO port. Possible values are in Table 206: GpioPort Enumeration Values
	1380	4	CubeWheel3 Enable Pin	ENUM	CubeWheel3 Enable GPIO port pin. Possible values are in Table 207: GpioPortPin Enumeration Values

Table 202: AcpProgramType Enumeration Values

Numeric Value	Name	Description
0	3-Axis ACP	3-Axis ACP
1	Y-Momentum ACP	Y-Momentum ACP

Table 203: SpecialConSelect Enumeration Values

Numeric Value	Name	Description
0	None	None
1	Z-axis Sun pointing	Z-axis Sun pointing
2	GEO Tracking	GEO Tracking
3	Inertial pointing	Inertial pointing
4	Y-axis Sun pointing	Y-axis Sun pointing

Table 204: CsCamType Enumeration Values

Numeric Value	Name	Description
0	Sun Sensor	Sun Sensor
1	Nadir Sensor	Nadir Sensor

Table 205: GpsSelect Enumeration Values

Numeric Value	Name	Description
0	No GPS Support	No GPS support
1	Novatel GPS	Novatel GPS
2	Skyfox GPS	Skyfox GPS

Table 206: GpioPort Enumeration Values

Numeric Value	Name	Description
0	Port A	Port A
1	Port B	Port B
2	Port C	Port C
3	Port D	Port D
4	Port E	Port E
5	Port F	Port F

Table 207: GpioPortPin Enumeration Values

Numeric Value	Name	Description
0	Pin 0	Pin 0
1	Pin 1	Pin 1
2	Pin 2	Pin 2
3	Pin 3	Pin 3
4	Pin 4	Pin 4
5	Pin 5	Pin 5
6	Pin 6	Pin 6
7	Pin 7	Pin 7
8	Pin 8	Pin 8
9	Pin 9	Pin 9
10	Pin 10	Pin 10
11	Pin 11	Pin 11
12	Pin 12	Pin 12
13	Pin 13	Pin 13

14	Pin 14	Pin 14
15	Pin 15	Pin 15

Table 208: User-coded Controller and Estimator Parameters Message Format

Set ID/Get ID	29/226		**Parameters Length (bytes)**		96
Description	Settings for user-coded estimation and control modes				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	384	User Coded Controller Settings	ARRAY	Settings for user coded control mode(s)
	384	384	User Coded Estimator Settings	ARRAY	Settings for user coded estimation mode(s)

Table 209: Augmented-SGP4 Parameters Message Format

Set ID/Get ID	28/227		**Parameters Length (bytes)**		30
Description	Settings for GPS augmented SGP4				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Incl. coefficient	UINT	Inclination filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	16	16	Raan coefficient	UINT	RAAN filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	32	16	Ecc coefficient	UINT	Eccentricity filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	48	16	Aop coefficient	UINT	Argument of perigee filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	64	16	Time coefficient	UINT	Time filter coefficient. Raw parameter value is obtained

					using the formula: (formatted value) = RAWVAL*0.001
	80	16	Pos coefficient	UINT	Position filter coefficient. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.001
	96	8	Maximum position error	UINT	Maximum position error for asgp4 to continue working. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.1
	104	8	ASGP4 filter	ENUM	The type of filter asgp4 is using. Possible values are in Table 193: AsgpFilter Enumeration Values
	112	32	xp coefficient	INT	Polar coefficient xp. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.0000001
	144	32	yp coefficient	INT	Polar coefficient yp. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.0000001
	176	8	GPS roll over	UINT	GPS roll over number
	184	8	Position sd	UINT	Maximum position standard deviation for asgp4 to operate. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.1
	192	8	Velocity sd	UINT	Maximum velocity standard deviation for asgp4 to operate. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	200	8	Min. satellites	UINT	Minimum satellites for asgp4 to operate
	208	8	Time gain	UINT	Time offset compensation gain. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL*0.01
	216	8	Max. lag	UINT	Maximum lagged timestamp measurements to incorporate. Raw parameter value is obtained

					using the formula: (formatted value) = RAWVAL*0.01
	224	16	Min. samples	UINT	Minimum samples to use for asgp4 process

Table 210: SD Log1 Configuration Message Format

Set ID/Get ID	104/235		**Parameters Length (bytes)**		13
Description	Log selection and period for LOG1				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	80	Log Selection	ARRAY	Log Selection – up to 80 flags indicating which telemetry frames should be logged
	80	16	Log Period	UINT	Log period. Set to 0 to disable logging
	96	8	Log Destination	ENUM	Which SD card to use to store log file. Possible values are in Table 211: SdLogSelect Enumeration Values

Table 211: SdLogSelect Enumeration Values

Numeric Value	Name	Description
0	Primary SD Card	Use Primary (on-board) SD card for log file
1	Secondary SD Card	Use Secondary SD card for log file

Table 212: SD Log2 Configuration Message Format

Set ID/Get ID	105/236		**Parameters Length (bytes)**		13
Description	Log selection and period for LOG2				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	80	Log Selection	ARRAY	Log Selection – up to 80 flags indicating which telemetry frames should be logged
	80	16	Log Period	UINT	Log period. Set to 0 to disable logging

	96	8	Log Destination	ENUM	Which SD card to use to store log file. Possible values are in Table 211: SdLogSelect Enumeration Values

Table 213: UART Log Configuration Message Format

Set ID/Get ID	106/237		**Parameters Length (bytes)**		12
Description	Log selection and period for UART (unsolicited TLM)				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	80	Log Selection	ARRAY	Log Selection – up to 80 flags indicating which telemetry frames should be logged
	80	16	Log Period	UINT	Log period. Set to 0 to disable logging

Table 214: Inertial Pointing Reference Vector Message Format

Set ID/Get ID	34/238		**Parameters Length (bytes)**		6
Description	Reference unit vector for inertial pointing control mode				
Parameters	**Offset (bits)**	**Length (bits)**	**Name**	**Data Type**	**Description**
	0	16	Inertial Reference X	INT	Inertial Reference X. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	16	16	Inertial Reference Y	INT	Inertial Reference Y. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL/10000.0
	32	16	Inertial Reference Z	INT	Inertial Reference Z. Raw parameter value is obtained using the formula: (formatted value) = RAWVAL/10000.0

6. Firmware Operational Sequences

This section describes sequences of commands and telemetry request that must be issued to perform higher level functions.

6.1 Bootloader and ACP (or other application) Identification

The suggested logic for an external OBC that interfaces with the CubeComputer and bootloader is to switch on the CubeComputer or ADCS, wait for 10s and then request the identification telemetry frame (TLM ID 128). This will inform the external OBC whether the application did indeed boot the ACP program (the node type field for this telemetry frame will differ for the bootloader and the application).

If the application failed to boot (i.e. the Bootloader is still active after 10s), other telemetry requests to the bootloader may follow to find out what the status is (failed boot attempts, error switching on external SRAM, or some other condition).

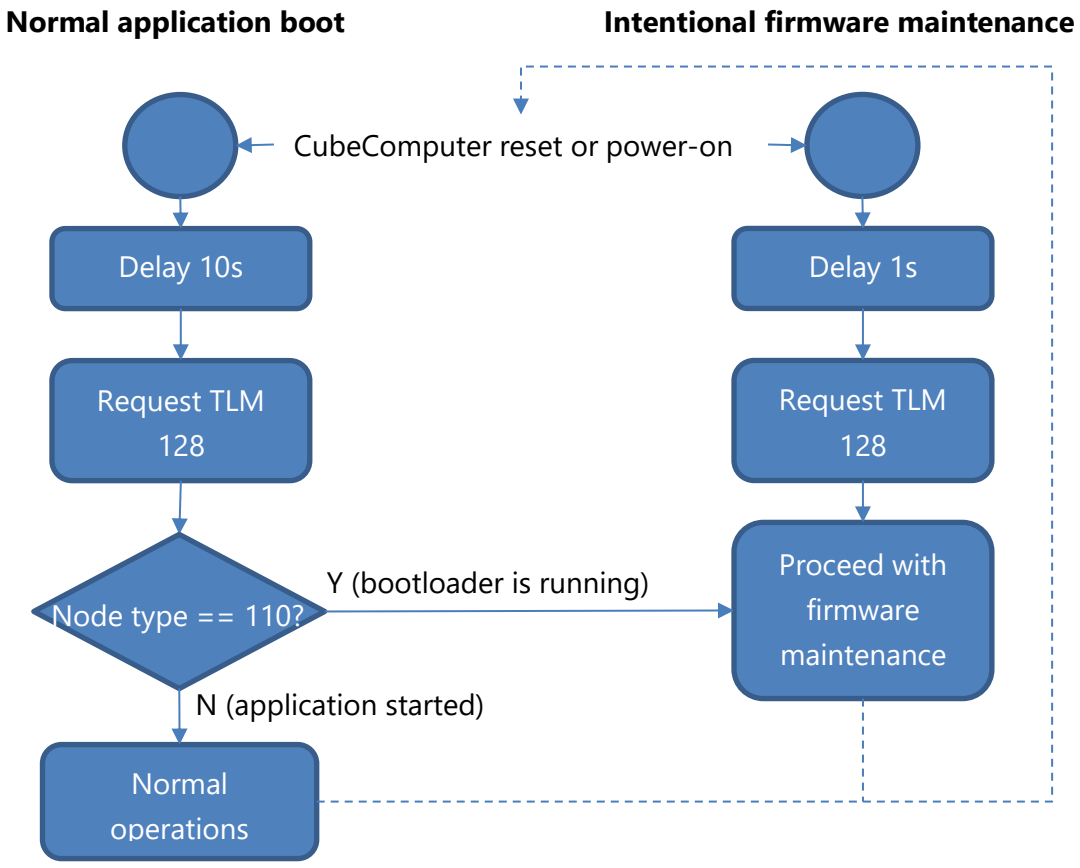


Figure 12: Example external OBC logic for Bootloader operation

6.2 SD Card File Management

Files that are stored on the SD card of CubeComputer can be downloaded and managed using the protocol described here.

6.2.1 Download file list

The current download-able list of files on the SD card can be found in two ways. The first method involves a sequence of *File Information* telemetry requests. But first, the file list read pointer must be reset with a *Reset File List Read Pointer* command (Table 24: Reset File List Read Pointer Command Format).

Following the *Reset File List Read Pointer* command, the next available file in the download list can be queried with the *File Information* telemetry request (Table 43: File Information Telemetry Format). The *File Information* telemetry frame should be polled until the *Busy Updating* flag is cleared. At this point the populated file information will be valid. If the populated file information comes back as all zeroes, the end of the file list has been reached. If not, the file list read pointer can be advanced to the next file with the *Advance File List Read Pointer* command (Table 18: Advance File List Read Pointer Command Format).

The complete process is described by the flow diagram in .

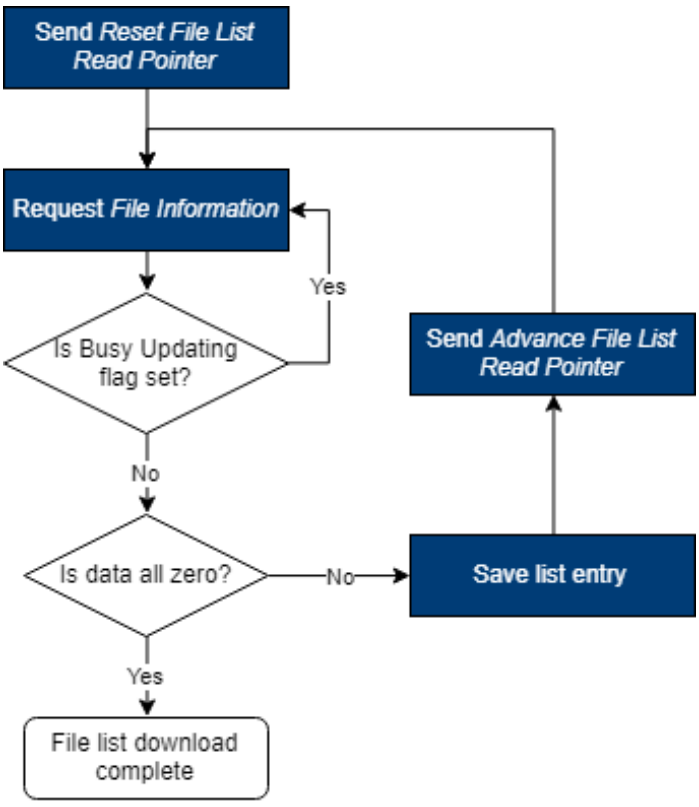


Figure 13 – File list download process.

The other method of retrieving the downloadable file list is to perform an actual file transfer. In this case the file selection (for the file to download) should be specified as *Index File*.

6.2.2 Index File Format

The Index File is a virtual file on the CubeComputer. It contains the list of downloadable files on the SD card with the file meta-data included. The downloaded file index will contain a 37-byte entry for every file on the SD card:

Table 215: Format of entries in the Index File

Byte offset	Field length (bytes)	Content
0	1	File Type enumeration
1	1	File counter
2	2	16-bit CRC checksum
4	4	File size (in bytes)
8	4	File creation date/time (MSDOS format)
12	25	File meta-data (structure of this depends on the file type)

The following code can be used to convert from the MSDOS time format to conventional date and time.

```
/*MS-DOS Date & Time (4 bytes)

The lower word determines the time, the upper word the date. Used by several DOS
function calls and by all FAT file systems.

Bits Contents
0-4 Second divided by 2
5-10 Minute (0-59)
11-15 Hour (0-23 on a 24-hour clock)
16-20 Day of the month (1-31)
21-24 Month (1 = January, 2 = February, etc.)
25-31 Year offset from 1980
*/

seconds = (msdostime & 0x1f) << 1;  // 5bits - 32
minutes = (msdostime >> 5) & 0x3f;  // 6buts - 64
hour = (msdostime >> 11) & 0x1f;    // 5bits - 32
day = (msdostime >> 16) & 0x1f;     // 5bits - 32
month = (msdostime >> 21) & 0x0f;   // 4bits - 16
year = (msdostime >> 25) + 1980;    // 7bits - 128 (1980 to 21..)
```

The file meta-data is dependent on the type of file. It is currently only used by telemetry log files, and in this case the first 10 bytes of the meta-data matches the telemetry mask that was specified when creating the log file.

6.2.3 File download

Files are downloaded from CubeComputer in blocks. A block is typically downloaded as a burst of packets, of which some packets may get lost. A "Hole Map" is used to keep track which packets in a block have been successfully received. The Hole Map is then communicated to the CubeComputer, and a burst retry is performed for all the missed packets.

The download of a file block is initiated by sending a *Load File Download Block* command (Table 17: Load File Download Block Command Format). This will read a block of data from the indicated file and buffer it in SRAM on the CubeComputer.

The file is uniquely identified by the *File Type* and *Counter* parameters. The *Offset* and *Block Length* parameters indicate which part of the file to buffer in memory. The maximum *Block Length* is 20 kB.

The *Download Block Ready* telemetry frame (Table 42: Download Block Ready Telemetry Format) can be polled to see if the block is done loading. Once the block is ready, the burst can be initiated by sending an *Initiate Download Burst* command (Table 25: Initiate Download Burst Command Format), with the *Ignore Hole Map* parameter set to true.

If the *Initiate Download Burst* command was sent on the UART, the CubeComputer application will respond by sending up to 1024 packets (the number depends on the *Block Length* specified with the *Load File Download Block* command), each with a payload length of 20 bytes in rapid succession. Each download packet will have a header ID that matches that of the *Initiate File Download Burst* command. The following two bytes will contain the counter of the packet in the burst. The counter makes it possible to keep track of which packets have been received on the remote side.

If the *Initiate File Download Burst* command is issued on the I^2C communications link, it is expected to successively perform up to 1024 read transactions from the remote side, each with 22 bytes length, again with the same format as the *Download File Packet* above.

The *Hole Map* is a bitmap, where each bit represents one 20-byte packet out of the complete download block. Since there are at most 1024 packets in one file download block, the *Hole Map* is also 1024 bits long, or 128 bytes.

To make the CubeComputer resend only the packets that have been missed by the remote side, the remote component must upload the Hole Map to the CubeComputer by performing up to 8 *Hole Map* commands (each *Hole Map* command contains 16 bytes of the Hole Map).

After uploading the Hole Map, another *Initiate Download Burst* command can be issued, but this time with the Ignore Hole Map parameter set to false. The CubeComputer will then transmit only the packets that have a corresponding '0' in the Hole Map.

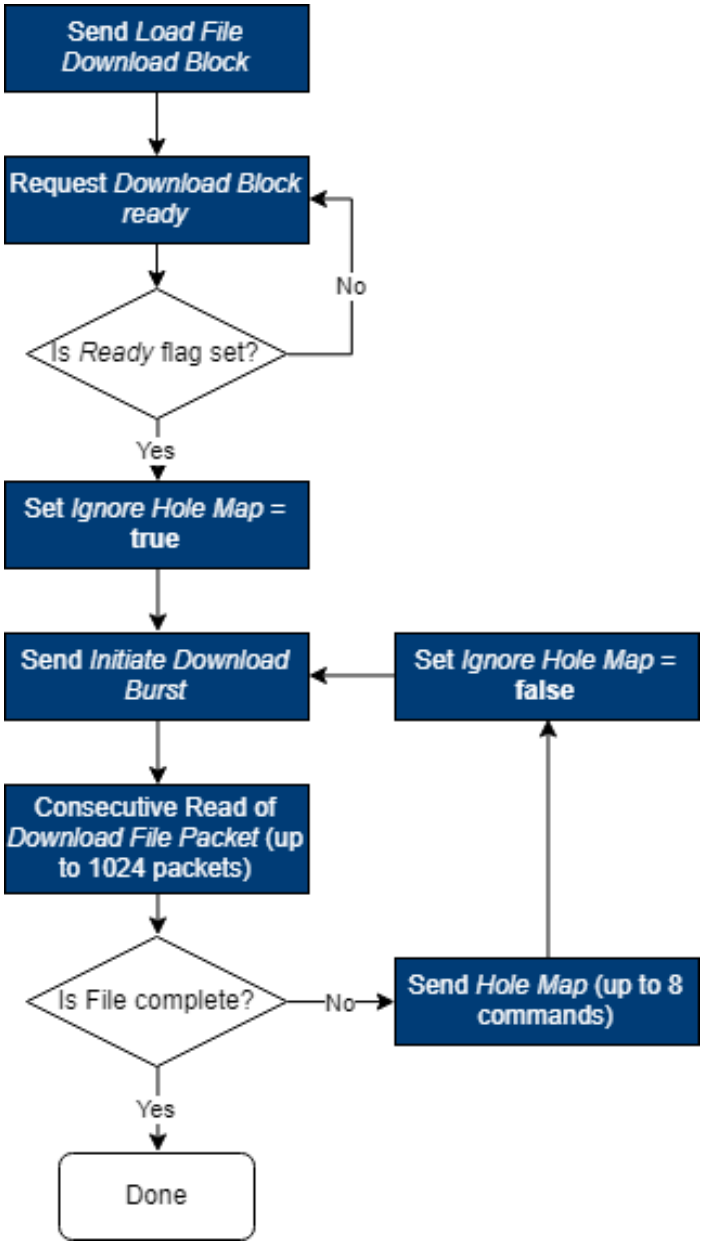


Figure 14: File download process.

6.2.4 File erase and SD card formatting

To delete files, use the *Erase File* command (Table 15: Erase File Command Format). If the *Erase All* flag in the *Erase File* command is set, all files on the SD card will be erased. This operation is not instantaneous and requires some time to complete. The status of the *Erase All* operation can be monitored using the *Erase All Busy* flag in the *SD card format/erase progress* telemetry frame (see Table 38: SD card format/erase progress Telemetry Format).

The SD card can also be formatted using the *Format SD card* command (see Table 14: Format SD card Command Format). This operation is not instantaneous and requires some time to complete. The status of the formatting operation can be monitored using the *Format Busy* flag

in the *SD card format/erase progress* telemetry frame (see Table 38: SD card format/erase progress Telemetry Format).

6.3 Retrieving Flash Firmware Information (Bootloader only)

The programs programmed into the flash memory at each program location in **Error! Reference source not found.**, and also its CRC checksum can be found by first sending a *Read Program Information* command (see Table 65: Read Program Information Command Format). The information is returned by reading the Program Information telemetry (see Table 70: Program Information Telemetry Format). It might take some time to calculate the CRC on the CubeComputer. The telemetry request can be polled, using the busy flag to test if it has completed. A timeout value of 2 seconds is suggested.

6.4 Firmware uploads (Bootloader only)

The process of firmware uploads makes use of block transfers of 20kB at a time with error checking and retries. It thus also allows for a communications link that receives occasional errors. The complete process is illustrated in Figure 15.

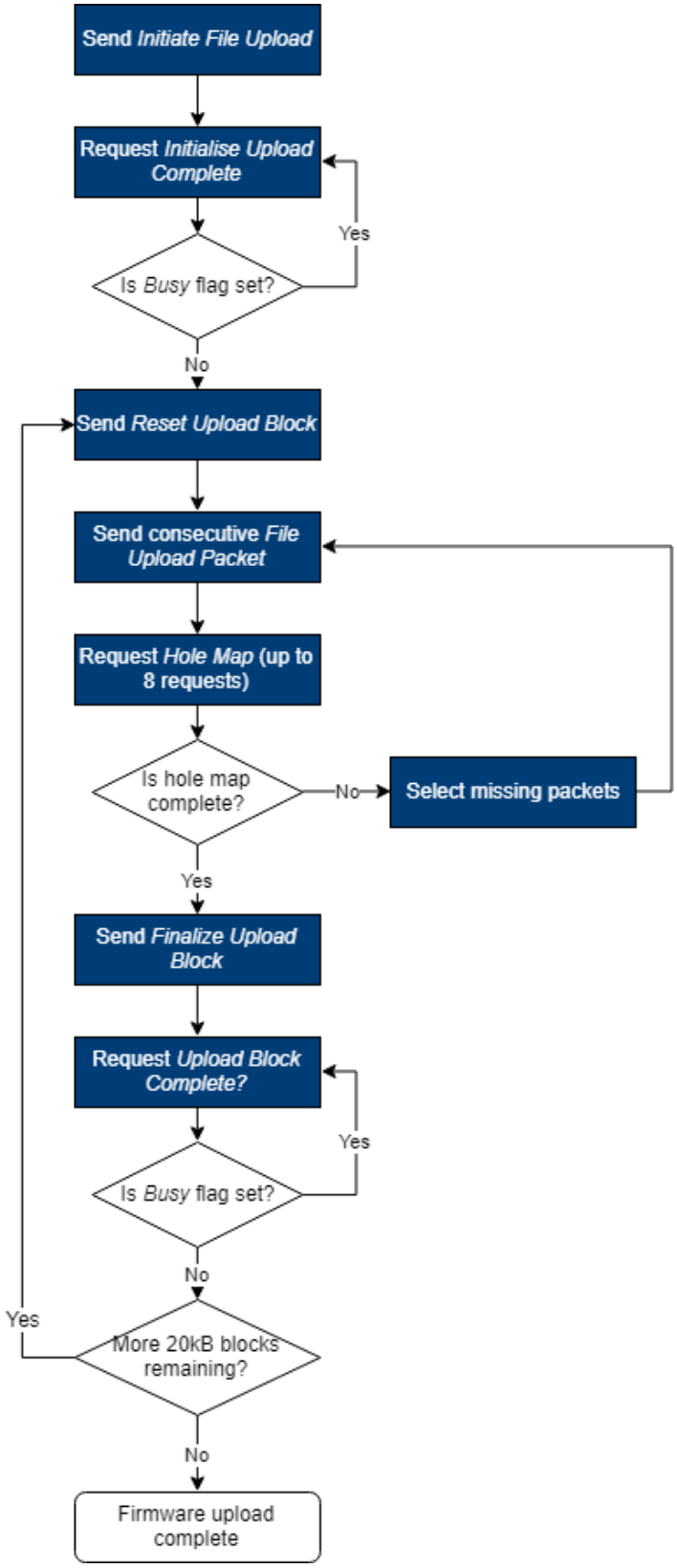


Figure 15: File upload process.

A new firmware file can only be uploaded to one of the external flash memory locations. Note that file upload destinations to the SD card, mentioned in the FileUploadDestination enumeration are **not supported**.

The process is initiated by first erasing the selected flash program area. This is done by sending an *Initiate File Upload* telecommand (see Table 19: Initiate File Upload Command Format). The blocksize parameter of this command is ignored and can be set to 0. It takes a while for the flash erase procedure to complete. The status can be polled with an *Initialize Upload Complete* telemetry request (see Table 44: Initialize Upload Complete Telemetry Format). It is suggested to wait up to 8s for the erase to complete.

File or firmware uploads now proceed in iterations of 20kB blocks.

At the start of a block transfer, the ADCS CubeComputer is notified of the block that is to be uploaded by sending a Reset Upload Block telecommand (see Table 23: Reset Upload Block Command Format). This command will reset the Hole Map.

The file or firmware data is then uploaded to the CubeComputer by sending op to 1024 messages, each with a size of 20 bytes. Each 20-byte file message is sent using the File Upload Packet telecommand (see Table 21: File Upload Packet Command Format). These commands are not acknowledged and may be sent in a continuous stream.

To find out if the CubeComputer received all the packets, the OBC must read the hole map from the ADCS CubeComputer. This is done using TLM IDs 247 – 254 (see Table 52: Hole Map 1 Message Format).

The OBC will then resent the packets that were missed as indicated by the hole map, using the following code:

```
for (int I = 0; I < 1024; i++)
{
    if ((holemap[I >> 3] & (1 << (I & 0x07))) == 0)
    {
        // missed!
    }
}
```

The holemap array in the example above is a 128-byte array, built-up by concatenating all the holemap fields from the Hole Map telemetry replies.

Once the entire 20kB block has been successfully uploaded, it can be committed to flash memory using the Finalize Upload Block command (Table 22: Finalize Upload Block Command Format). This command takes as parameter the flash memory program slot – valid values are 3 to 9 (Flash program 1 to 7), the offset into the program slot, and the length of valid data in the 20kB buffer.

The flash programming process can be polled by requesting the Upload Block Complete telemetry (Table 45: Upload Block Complete Telemetry Format) and checking the Busy flag. A suitable timeout for this process is 10s.

The upload stops when all the file blocks have been uploaded and programmed successfully.

A final CRC check should be performed on the completed flash program. The firmware file CRC on the CubeComputer is found by the process described in 6.3.

6.4.1 CRC Checksum

File transfers make use of a CRC16 checksum. The following code can be used to compute the checksum of a buffer or file.

```
Uint16_t CRC_Calc(uint8_t *start, uint32_t len)
{
  uint16_t crc = 0;
  uint8_t  *data;
  uint8_t  *end = start + len;

  for (data = start; data < end; data++)
  {
    crc  = (crc >> 8) | (crc << 8);
    crc ^= *data;
    crc ^= (crc & 0xff) >> 4;
    crc ^= crc << 12;
    crc ^= (crc & 0xff) << 5;
  }
  return crc;
}
```

7. Telemetry logging (CubeACP only)

The CubeACP can save telemetry frames to a log file on the SD card, and output "unsolicited" telemetry on the UART. In both cases, the telemetry frames that are selected for logging, are specified using a 10-byte mask. The bits in the mask each represent a specific telemetry frame (with specific ID). Not all telemetry frames can be logged. The frames which can be logged, their IDs and respective bit-position in the 10-byte mask can be found in Table 216: Lo. All loggable telemetry frames are 6 bytes in length.

Table 216: Loggable Telemetry Frames

Mask	Telemetry ID	Telemetry Frame
01 00 00 00 00 00 00 00 00 00	144	Communication Status
02 00 00 00 00 00 00 00 00 00	143	EDAC Error Counters
08 00 00 00 00 00 00 00 00 00	141	Last Logged Event
10 00 00 00 00 00 00 00 00 00	142	SRAM Latchup counters
80 00 00 00 00 00 00 00 00 00	157	Magnetorquer Command
00 01 00 00 00 00 00 00 00 00	158	Wheel Speed Commands
00 02 00 00 00 00 00 00 00 00	151	Magnetic Field Vector
00 04 00 00 00 00 00 00 00 00	152	Coarse Sun Vector
00 08 00 00 00 00 00 00 00 00	153	Fine Sun Vector
00 10 00 00 00 00 00 00 00 00	154	Nadir Vector
00 20 00 00 00 00 00 00 00 00	155	Rate Sensor Rates
00 40 00 00 00 00 00 00 00 00	156	Wheel Speed
00 80 00 00 00 00 00 00 00 00	181	Star 1 Body Vector
00 00 01 00 00 00 00 00 00 00	184	Star 1 Orbit Vector
00 00 02 00 00 00 00 00 00 00	182	Star 2 Body Vector
00 00 04 00 00 00 00 00 00 00	185	Star 2 Orbit Vector
00 00 08 00 00 00 00 00 00 00	183	Star 3 Body Vector
00 00 10 00 00 00 00 00 00 00	186	Star 3 Orbit Vector
00 00 20 00 00 00 00 00 00 00	171	CubeSense1 Current Measurements
00 00 40 00 00 00 00 00 00 00	172	CubeControl Current Measurements
00 00 80 00 00 00 00 00 00 00	173	Wheel Currents
00 00 00 01 00 00 00 00 00 00	198	ADCS Misc Current Measurements
00 00 00 02 00 00 00 00 00 00	174	ADCS Temperatures
00 00 00 04 00 00 00 00 00 00	175	Rate sensor temperatures
00 00 00 08 00 00 00 00 00 00	132	Current ADCS State
00 00 00 10 00 00 00 00 00 00	146	Estimated Attitude Angles
00 00 00 20 00 00 00 00 00 00	218	Estimated Quaternion
00 00 00 40 00 00 00 00 00 00	147	Estimated Angular Rates
00 00 00 80 00 00 00 00 00 00	148	Satellite Position (ECI)

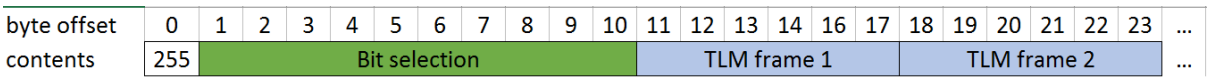
00 00 00 00 01 00 00 00 00 00	149	Satellite Velocity (ECI)
00 00 00 00 02 00 00 00 00 00	150	Satellite Position (LLH)
00 00 00 00 04 00 00 00 00 00	219	ECEF Position
00 00 00 00 08 00 00 00 00 00	159	IGRF Modelled Magnetic Field Vector
00 00 00 00 10 00 00 00 00 00	160	Modelled Sun Vector
00 00 00 00 20 00 00 00 00 00	161	Estimated Gyro Bias
00 00 00 00 40 00 00 00 00 00	162	Estimation Innovation Vector
00 00 00 00 80 00 00 00 00 00	163	Quaternion Error Vector
00 00 00 00 00 01 00 00 00 00	164	Quaternion Covariance
00 00 00 00 00 02 00 00 00 00	165	Angular Rate Covariance
00 00 00 00 00 04 00 00 00 00	176	Raw GPS Status
00 00 00 00 00 08 00 00 00 00	177	Raw GPS Time
00 00 00 00 00 10 00 00 00 00	178	Raw GPS X
00 00 00 00 00 20 00 00 00 00	179	Raw GPS Y
00 00 00 00 00 40 00 00 00 00	180	Raw GPS Z
00 00 00 00 00 80 00 00 00 00	166	Raw Cam2 Sensor
00 00 00 00 00 00 01 00 00 00	167	Raw Cam1 Sensor
00 00 00 00 00 00 02 00 00 00	168	Raw CSS 1 to 6
00 00 00 00 00 00 04 00 00 00	169	Raw CSS 7 to 10
00 00 00 00 00 00 08 00 00 00	170	Raw Magnetometer
00 00 00 00 00 00 10 00 00 00	188	Star Performance1
00 00 00 00 00 00 20 00 00 00	187	Star Magnitude
00 00 00 00 00 00 40 00 00 00	212	Star 1 Raw Data
00 00 00 00 00 00 80 00 00 00	213	Star 2 Raw Data
00 00 00 00 00 00 00 01 00 00	214	Star 3 Raw Data
00 00 00 00 00 00 00 02 00 00	189	Star Timing
00 00 00 00 00 00 00 04 00 00	215	Secondary Magnetometer Raw Measurements
00 00 00 00 00 00 00 08 00 00	201	Fine Estimated Angular Rates
00 00 00 00 00 00 00 10 00 00	216	Raw Rate Sensor
00 00 00 00 00 00 00 20 00 00	224	Current ADCS State 2
00 00 00 00 00 00 00 40 00 00	229	CubeStar Estimated Rates
00 00 00 00 00 00 00 80 00 00	230	CubeStar Estimated Quaternion
00 00 00 00 00 00 00 00 01 00	231	Star Performance2
00 00 00 00 00 00 00 00 02 00	232	CubeSense2 Current Measurements

7.1 Telemetry log file format

The telemetry log files that are created on the SD card has a 10-byte header, which matches with the mask that was specified when the log was initiated. Following this, there will be *M* log entries, where each log entry starts with the Unix time (32-bit unsigned integer) and milliseconds (16-bit unsigned integer) (6 bytes total). The rest of the log entry is filled with *N* telemetry frames, where N is the number of bits that have been set in the 10-byte mask. The size of the log file will then equal 10+6*(N+1)*M.

7.2 UART Unsolicited telemetry log

Telemetry log frames which are output on the UART have the following format. The message identifier is always 255 (0xFF). This is followed by the 10-byte mask which further determines which telemetry frames follow.



Content	Log ID (0xFF)	Mask	TLM frame 1	TLM frame 2	...	TLM frame N
Offset (bytes)	0	1	11	17	...	11 + (N-1)*6
Len (bytes)	1	10	6	6	...	6

Figure 16: Contents of UART log unsolicited telemetry transmissions

The total length of the message is 11+N*6 bytes where N is the number of bits that are set in the mask.

8. Document History

Version	Responsible person(s)	Pages	Date	Description of change
3.14	GJVV	All	18/05/2020	Updates for ACP v7 and Bootloader v3: - Reference document version numbers - TC/TLM tables (autogen) updated for ACP and bootloader - Added SD card formatting details - Some minor language edits
7.1	GJVV	2	18/05/2020	New document version numbering scheme – major matches ACP interface version
7.2	GJVV	161	03/11/2020	Removed driver library section
7.3	LV	106	30/3/2021	Added +XY 45 degree option to AxisSelect enumeration (Table 180)
7.4	LV	159	5/1/2022	Removed outdated/unnecessary reference to CRC telemetry request in firmware upload text, and updated process description.