Title

AS250 Generic

Generic TM/TC
Interface Control Document

	Name and Function	Date	Signature
Prepared by	**D.BISCARROS** *(AS250 CSW Architect)*	30/05/11	
Verified by	**Philippe LEBLOND** *(AS250 Operations)* **Hervé MARCILLE** *(AS250 Avionics Technical Synthesis)*	30/05/11 31/05/11	Ph. LEBLOND
Approved by	**S.CAZALS** *(AS250 Product Assurance)*	31-05-11	
Authorized by	**Ange DEFENDINI** *(AS250 Development Manager)*	07/06/11	
Application authorized by			

Document type	Nb WBS	Keywords
		ICD, Generic, TM/TC

SUMMARY

This document defines the telemetry and telecommand structures and the PUS services for the ASTROSAT-250 (AS250) missions.

DOCUMENT CHANGE LOG

Issue/ Revision	Date	Modification Nb	Modified pages	Observations
01/05	17/06/10		§5.1.2.2	Correction according to SEOSAT PDR RID SP-102
			§7.20	Correction of service 140 description
			Tables 7.9-B, 7.17-A	Correction of typos
			§7.6	SAU for DMS fixed to 8 bits
			§7.5, 7.12	Event severity identified by the EID in service 12
01/06	23/11/2010		Tables 8.2-B and 7.3-1	Correction of TM(3,26) category value
			§7.6	Obsolete reference to table 8.6-1 removed (CSW1-ASF-009)
			Table 7.20-A	Title corrected (CSW1-ASF-013)
			§4.3.3.6 and 4.3.3.9	CSW1-ASF-065: Parameter type (7, 0) not allowed in TM packets. Parameter type (11,0) allowed only in TM packets.
			§4.3.3.2	Enums values are unsigned (SCOS2000 constraint)
			§4.3.3.8	New type for coarse time parameters
			Table 1.1-A	Correction of ground source ID
01/07	05/05/2011		§6.1.3	Correction of CPDU packet description
			§7.18	Service 18 extension (DIV.CR.00081.T.ASTR)

PAGE ISSUE RECORD

Issue of this document comprises the following pages at the issue shown

Page	Issue/ Rev.	Page	Issue/ Rev.	Page	Issue/ Rev.	Page	Issue/ Rev.	Page	Issue/ Rev.	Page	Issue/ Rev.
All	01/07										

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

1 INTRODUCTION AND SCOPE

1.1 SCOPE

This document defines the relevant TM/TC parameters for the interface between ASTROSAT-250 spacecrafts and ground segments, as well as the list of other documents applicable to this interface.

This document serves to complement and extend the Packet Telemetry and Telecommand Standards (AD-01, AD-02, AD-03 and AD-04) by defining all the mission specific details of the various interface levels, and in addition the Application-Level interface between ground and on-board applications.

This document describes the data structures for telemetry and telecommand packets implemented to support the operational requirements of the ASTROSAT-250 missions. Being tailored especially for the AS250 missions, this document supersedes the ESA Packet Utilization Standard (PUS) from which it is derived.

1.2 OPERATIONS SCENARIO

1.2.1 Nominal Mission Operations

The mission operations of the ASTROSAT-250 spacecrafts and their payloads will be conducted under control of Mission Operations Centres. This task includes spacecraft operations required for all phases of the mission.

Throughout the complete mission duration (from launch up to the end of mission, when ground contact to the spacecraft/payload is terminated), facilities and services will be provided to the scientific community for scientific data acquisition.

Interaction with the spacecraft will be by monitoring and analysis of telemetered data and the uplink of telecommands to effect the necessary operations. Most telecommands will be stored on board for later execution at a defined time, others may be intended for execution in «near-real-time». In both cases, it may also be necessary to control subsystem and experiment equipment using low-level commands or high-level commands (i.e. via on-board applications or On-Board Control Procedures).

Telemetry and telecommands will also be required for on-board software management functions, including:

- control of, and communication with, on-board processes (such as an on-board telemetry monitor)
- loading and dumping of on-board memories
- control of on-board Mission Time Line.

All telecommands must be appropriately verified in telemetry at acceptance and execution.

Telemetry data will be required in order to verify the execution of all mission operations and will also be required for:

- routine on ground health monitoring of the subsystems and the experiments;
- reporting to the ground any anomalous events detected on-board and any actions taken autonomously by the on-board systems;
- performance evaluation on the ground for the purposes of long-term trend analysis and feedback into the mission planning cycle.

1.2.2 Contingency Operations

In case of unforeseen on-board events, actions will be necessary to investigate and correct anomalies utilising the available telemetry and command functions. In addition, it may be necessary to modify on-board software in order to compensate for on-board failures or anomalous performance.

1.2.3 Packet Distribution

The following telemetry and telecommand packet categories exist :

- those generated on the ground and uplinked to the spacecraft (TC packets)
- those generated by on board applications and downlinked to the ground (TM packets)

There is no packetized end users on ASTROSAT-250 so that:

- TC packets from ground are decoded by the OBC software and the data content routed to the user, either directly or after storage in PUS services 11, 18 and 19,
- TM data is acquired by the OBC software on discrete lines and packetized by the OBC software for downlink to the ground, or on-board analysis, or storage.

An exception to the above exists in the case of high priority CPDU commands which are decoded by the CPDU and routed directly to end users as discrete (pulse) commands. CPDU commands have one dedicated Process ID.

1.3 PUS TAILORING

In this tailored version of the PUS document, following changes have been made to standard services:

- PUS service 12: check definitions are associated to a monitoring ID instead of a parameter ID
- PUS services 14 & 15: layout of TM reports have been changed to accommodate the complex selection criterias.

2 APPLICABLE AND REFERENCE DOCUMENTS

2.1 APPLICABLE DOCUMENTS

The following list of documents contains documents made applicable for establishing this document. Unless an issue is quoted for a document, the current issue is deemed to apply. When an issue is quoted, that issue and no other must be used.

AD-01	Telemetry transfer frame protocol	ECSS-E-ST-50-03C
AD-02	TC protocols synchronization and channel coding	ECSS-E-ST-50-04C
AD-03	Telemetry and telecommand Packet Utilization	ECSS-E-70-41A
AD-04	TM synchronization and channel coding	ECSS-E-ST-50-01C

2.2 REFERENCE DOCUMENTS

The following list of documents contains documents given in references. Unless an issue is quoted for a document, the current issue is deemed to apply. When an issue is quoted, that issue and no other must be used.

RD-01	The Application of CCSDS protocols to Secure Systems	CCSDS 250.0-G-2
RD-02	Telecommand Decoder Specification	PSS-04-151
RD-03	AS250 OSCAR User Manual	OSCAR.MU.DA0024626.V.ASTR
RD-04	ASIC SCOC3 User's Manual	R&D.SCOC3.NT.00660.V.ASTR
RD-05	AS250 MIL-STD-1553B bus protocol specification	DIV.SP.00030.T.ASTR

3 ACRONYMS LIST AND GLOSSARY OF TERMS

3.1 ACRONYMS LIST

A

AOCS	Attitude and Orbit Control System
APID	Application Identifier
ARO	Automatic Reconfiguration Order
AVB	Avionic Bus

B

BC	Bus Controller (MIL-STD-1553B bus)

C

CPDU	Command Pulse Distribution Unit
CRC	Cyclic Redundancy Code
CSW	Central SoftWare
CUC	CCSDS Unsegmented Time Code

D

DHS	Data Handling System
DMS	Data Management System
DTC	Direct TeleCommand

E

EEPROM	Electrically Erasable Programmable Read-Only Memory
EID	Event Identification

F

FAR	Frame Analysis Report
FDIR	Failure Detection, Isolation and Recovery
FID	Failure Identification

G

Gbit	109 bits
Gibit	230 bits

H

HK	HouseKeeping (telemetry data)
HPC	High Priority Command
HW	Hardware (also H/W)
I	
ICD	Interface Control Document
IF	Interface (also I/F)
L	
LCL	Latch Current Limiter
LLS	Low Level Signal
M	
MAP	Multiplexed Access Point
MLC	Memory Load Command
MM	Mass Memory
N	
NRZ	Non Return to Zero
O	
OBSR	On Board Storage and Retrieval
OBC	On-Board Computer
OBT	On Board Time
P	
PDEC	Packet Telecommand Decoder
PEC	Packet Error Control
PFC	Parameter Format Code
PID	Process Identifier
PLB	Payload Bus
PLDM	PayLoad Data Management
PM	Processor Module
PPS	Pulse Per Second
PROM	Programmable Read-Only Memory
PS	Packet Store

PUS	Packet Utilization Standard
R	
Red	Redundant
RAM	Random Access Memory
RT	Remote Terminal (MIL-STD-1553 bus)
RTC	Real Time Clock
RU	Reconfiguration Unit
S	
SA	Sub-Address (MIL-STD-1553 bus)
SCET	SpaceCraft Elapsed Time (time reference)
SDB	System DataBase
SEU	Single Event Upset
SGM	SafeGuard Memory
SID	Structure Identification
SM	Service Module
SpW	SpaceWire link (ECSS-E-50-12A)
SRAM	Static RAM
SW	Software (also S/W)
T	
TC	Telecommand
TFG	Transfer Frame Generator
TM	Telemetry
U	
UART	Universal Asynchronous Receiver Transmitter
URD	User Requirements Document
UTC	Universal Time Code
V	
VC	Virtual Channel
W	
WD	Watch-Dog

3.2 GLOSSARY OF TERMS

Application process	A continuous series of actions to bring about a result for a user. Such process may be on-board or on ground. Usually an application process can be associated with a subsystem or instrument. An application process can receive TC packets and/or generate TM packets.
Application data	Data destined to an on-board application process, encapsulated in a TC packet.
Application Process ID	An 11 bit address field. The application process ID is divided into two fields Process ID and Packet Category The PID of a TM packet identifies the application process which generated the packet. The PID of a TC packet identifies the application process which will receive the packet. An APID is unique across the system.
Channel	Physical input or output line(s).
Cycle	Execution time period of an application process.
(Functional) parameter	Variable that controls the result of a command, task or process.
FID	Function identifier, identifies a function of a task and defines the structure of the parameter field in the packet. The same FID may be used by different APIDs.
Packetised end user	On-board user which decodes TC packets and encodes TM packets. A packetised end user may have more than one application process.
MID	Memory identifier, identifies a memory within an application. The same MID may be used by different APIDs.
Non packetised end user	On-board user which does not decode TC packets or encode TM packets.
Parameter ID	Identifier that uniquely identifies a parameter across the system.
Process	See application process.
Register	A set of binary memory cells, fixed by design, to which data can be written and/or data can be read from.
SID	Structure identifier, defines the structure of the parameter field in the packet. The same SID may be used by different APIDs.
Source Data	Data generated by an on-board application process, encapsulated in a TM packet.
Task	A definite amount of actions to bring about a result for a user. One or more tasks may be active simultaneously within a process.

4 CONVENTIONS

4.1 BIT NUMBERING CONVENTIONS

The following convention shall be used to identify each bit in an N-bit field :

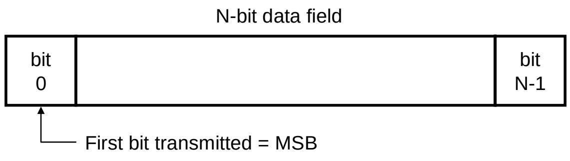


Figure 4.1-A : Bit Numbering

This convention is applicable to TM and TC. For 1553 bit numbering convention, which is different, refer to [RD-05].

1) The first bit in the field (starting from the left) is defined to be "Bit 0" and will be represented as the left most justified bit in a figure. The next bit is called "Bit 1", and so on, up to "Bit N-1", the bits being represented in this order from left to right in a figure.
2) If the N-Bit field is to be interpreted as "Unsigned Integer" value, Bit 0 is the MSB and Bit N-1 is the LSB.
3) If the N-Bit field is to be interpreted as "Signed Integer" value, Bit 0 indicates the sign with Bit 0 = 0 corresponding to a positive number and Bit 0 = 1 corresponding to a negative number.
4) Adjacent groups of bits are described in terms of bytes and words.
5) Byte = 1 byte = 8 bits (1 word = 2 bytes = 16 bits).
6) For multiple-byte words, the byte orientation is the same as the bit orientation. The first byte in the word (starting from the left) is defined to be "Byte 0", is the Most Significant Byte and is transmitted first. The next byte is called "Byte 1", and so on, up to "Byte N-1".

4.2 FIELD ALIGNMENT CONVENTIONS

The following convention shall be used to construct packet parameter fields:

1) Parameters with a length longer or equal 16 bits shall be word aligned, i.e. the LSB shall coincide with the word boundary.
2) Parameters with a length shorter than 16 bits shall not be allowed to span over word boundaries.
3) Parameters with a length shorter than 16 bits shall be right-adjusted within the occupied 16-bit word, leaving any required padding-bits in the most significant bits of the 16-bit word.
4) If more than one parameter is held in a single word the parameters shall be right adjusted.

4.3 PARAMETER TYPES AND STRUCTURES

4.3.1 Introduction

Each field in a telecommand or telemetry packet described in this document is designed to hold a parameter value. Each parameter field has a type, defining the set of values that can be assigned to the parameter. The ASTROSAT-250 parameter types are defined below.

This chapter defines the physical encoding rules for each type, i.e. the permitted lengths of the parameter fields and the internal format used to encode values. This chapter does not define the conversion of data parameters into physical or engineering units or user messages.

When defining telecommand and telemetry packets only parameter types defined in this document shall be allowed.

4.3.2 Encoding formats of parameter types

The parameter type defines the range of possible parameter values. A given parameter type can vary in format and length. Each combination of parameter type and encoding format has an associated parameter code, which defines the type and its physical encoding.

The parameter code shall be used whenever a definition of a parameter field is required. The parameter codes shall be applicable to both telecommand and telemetry data.

The parameter code (PTC,PFC) is defined as follows:

Parameter Type Code (PTC)	Parameter Format Code (PFC)
Enumerated	Enumerated

The parameter code is written as (PTC, PFC) in the tables below.

4.3.3 Parameter type definitions

4.3.3.1 Boolean

Parameter Type	PTC	PFC	Length	Value/Range
Boolean	1	0	1 bit	0 = false, 1 = true

4.3.3.2 Enumerated Parameter

Enumerated parameters are parameters with distinct unsigned integer values only involved in logical operations (as opposed to numeric operations). The values that such a parameter can take are discrete and un-ordered. An error code is a typical example.

Parameter Type	PTC	PFC	Length
Enumerated Parameter	2	1	1 bit
	2	2	2 bits
	2	3	3 bits
	2	4	4 bits
	2	5	5 bits
	2	6	6 bits
	2	7	7 bits
	2	8	8 bits
	2	12	12 bits
	2	16	16 bits
	2	32	32 bits

4.3.3.3 Unsigned Integer

Parameter Type	PTC	PFC	Length	Value/Range
Unsigned Integer	3	0	4 bits	{0...15}
	3	1	5 bits	{0...31}
	3	2	6 bits	{0...63}
	3	3	7 bits	{0...127}
	3	4	8 bits	{0...255}
	3	5	9 bits	{0...511}
	3	6	10 bits	{0...1023}
	3	7	11 bits	{0...2047}
	3	8	12 bits	{0...4095}
	3	9	13 bits	{0...8191}
	3	10	14 bits	{0...16383}
	3	11	15 bits	{0...32767}
	3	12	2 bytes	{0...65535}
	3	13	3 bytes	{0...2exp24 - 1}
	3	14	4 bytes	{0...2exp32 - 1}

4.3.3.4 Signed Integer

Parameter Type	PTC	PFC	Length	Value/Range
Signed Integer	4	0	4 bits	{-8...7}
	4	1	5 bits	{-16...15}
	4	2	6 bits	{-32...31}
	4	3	7 bits	{-64...63}
	4	4	8 bits	{-128...127}
	4	5	9 bits	{-256...255}
	4	6	10 bits	{-512...511}
	4	7	11 bits	{-1024...1023}
	4	8	12 bits	{-2048...2047}
	4	9	13 bits	{-4096...4095}
	4	10	14 bits	{-8192...8191}
	4	11	15 bits	{-16384...16383}
	4	12	2 bytes	{-32768...32767}
	4	13	3 bytes	{-2exp23...2exp23 - 1}
	4	14	4 bytes	{-2exp31...2exp31 - 1}

4.3.3.5 Real

Parameter Type	PTC	PFC	Length	Sign	Exponent	Fraction
Real	5	1	4 bytes	bit 0	bit 1 - bit 8	bit 9 - bit 31
	5	2	8 bytes	bit 0	bit 1 - bit 11	bit 12 - bit 63

Two formats for real numbers shall be allowed:

PC(5,1):32-bit single floating format according to ANSI/IEEE Std 754-1985.

PC(5,2) 64-bit double floating format according to ANSI/IEEE Std 754-1985

4.3.3.6 String

Parameter Type	PTC	PFC	Length	Value/Range
Byte string	7	0	var.	Variable-length byte-string. (Allowed for TC only)
Byte string	7	> 0	PFC bytes	Fixed-length byte-strings.

4.3.3.7 Character-string parameter

Parameter Type	PTC	PFC	Length	Value/Range
Character string	8	> 0	PFC bytes	Fixed-length character-strings.

4.3.3.8 Time

Parameter Type	PTC	PFC	Length	Coarse	Fine	Format
Absolute time	9	15	4 bytes	4	0	CUC format
	9	17	6 bytes	4	2	CUC format
	9	18	7 bytes	4	3	CUC format

The CUC format is made of two fields:

- Coarse part: number of seconds
- Fine part: number of sub-seconds in power of 2 (LSB of 2-16 with a fine part of 2 bytes or 2-24 with 3 bytes)

4.3.3.9 Deduced parameter

Deduced parameters (PTC = 11, PFC = 0) is allowed for telemetry packets only: type and length of parameters are deduced from the value of the preceding parameter.

4.3.4 Packets data field structure

The data field of the AS250 TC and TM packets shall respect the structure rules set # 1 defined in section 23.6.2 of [AD-03] with the following limitations:

- Deduced structures shall not be used (i.e. only deduced parameters are allowed),
- Parameters shall be aligned on bytes boundaries.

4.3.5 Parameter identifiers

AS250 housekeeping parameters used in PUS services 3, 12 or 140 have the following structure:

Parameter ID (4 bytes)		
Filler	Process ID	Local ID
	Enumerated	Integer
1 bit	7 bits	24 bits

4.4 PACKET NUMBERING CONVENTIONS

Packet class and function is provided by packet type and packet subtype, included in the data field header of the packet.

The Packet Type numbering scheme is devised to provide correlation between TC packets and the resulting TM packets and is therefore non-contiguous : there are cases where for a certain TC type, there is no corresponding TM type. Appendix 1 provides a complete cross-reference table down to sub-type level.

To make identification simpler, service type and subtype are represented by two numbers, separated by a comma. The notations TC(a,b) or TM(a,b) for telecommands and telemetry describes TC or TM of Service Type "a" and Subtype "b", for example : TM (1,2) is a telemetry packet type 1, subtype 2, and TC (2,1) is a telecommand packet type 2, subtype 1. Subtype numbers within a service type shall be unique.

5 TELEMETRY STRUCTURE

5.1 TELEMETRY SOURCE PACKET

All telemetry source packets must conform to the structure defined in [AD-03] and tailored for AS250 in Figure 5.1 1 Telemetry Source Packet Fields below.

<table>
<tr><th colspan="8">PACKET PRIMARY HEADER (48 bits)</th><th colspan="3">PACKET DATA FIELD
(VARIABLE)</th></tr>
<tr><td rowspan="2">VERSION NUMBER</td><td colspan="4">PACKET IDENTIFICATION</td><td colspan="2">PACKET SEQUENCE CONTROL</td><td rowspan="2">PACKET DATA LENGTH</td><td rowspan="2">PACKET SECONDARY HEADER</td><td rowspan="2">SOURCE DATA</td><td rowspan="2">PACKET ERROR CONTROL</td></tr>
<tr><td>TYPE INDICATOR</td><td>PCKT. SECONDARY HEADER FLAG</td><td colspan="2">APPLICATION PROCESS ID</td><td>GROUPING FLAGS</td><td>SOURCE SEQUENCE COUNT</td></tr>
<tr><td>000</td><td>0</td><td></td><td>PID</td><td>PCAT</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>3 Bits</td><td>1 Bit</td><td>1 Bit</td><td>7 Bits</td><td>4 Bits</td><td>2 Bits</td><td>14 Bits</td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="5">2 bytes</td><td colspan="2">2 bytes</td><td>2 bytes</td><td>10 bytes</td><td>Variable</td><td>2 bytes</td></tr>
<tr><td colspan="8"></td><td colspan="3"><-- 1 to 4090 bytes for OBC Packets -></td></tr>
<tr><td colspan="11"><-- 4096 bytes maximum for Housekeeping TM source Packets --></td></tr>
</table>

Figure 5.1-A : Telemetry Source Packet Fields

5.1.1 Packet Primary Header

5.1.1.1 Version Number

The Version Number must be set to '000' BIN for all telemetry issued on-board. The ground segment shall reject with an alarm any packet received with a version number other than zero.

5.1.1.2 Packet Identification

5.1.1.2.1 Type Indicator

For telemetry source packets, the type must be set to zero.

5.1.1.2.2 Secondary Header Flag

This indicates the presence or absence of a Data Field Header and must be set to 1 except for Standard Spacecraft Time Packet and for Idle Packets where it is set to 0 (see Service 9 sub type 2 and Appendix 3 and Appendix 4).

5.1.1.2.3 Application Process ID (APID)

The Application Process ID uniquely identifies the on board source of the packet.

The application ID (APID) is structured into two fields :

- The Process ID (PID) which defines the application which is the source of the telemetry packet.
- The category which identifies different categories of packets to be processed by the addressee.

The choice of Application Process ID values (PIDs and categories) across the spacecraft subsystems and experiments are given in Appendix 2.

Two Application Process ID's have been reserved for special purposes, namely the Standard Spacecraft Time Source Packet and the Idle Packet.

APID = 0 is reserved for the time packet

APID = "11111111111" is reserved for idle packets.

Their use and data structure are provided in Appendix 3 and Appendix 4 respectively.

5.1.1.3 Packet Sequence Control

5.1.1.3.1 Grouping Flags

The grouping flags shall be used when a number of telemetry source packets originating from the same application process are sent in a group. The interpretation of the grouping flags shall be:

— 01 BIN means first packet of a group of packets;

— 00 BIN means continuation packet;

— 10 BIN means last packet of a group of packets;

— 11 BIN means "stand-alone" packet.

The grouping flags are available for Dump but also for any request to downlink onboard data structures: Service 3, TM housekeeping definition, Service 11, MTL content, S12, monitoring description, S14, S15, S19,…).

Each packet structure, even if part of a group, shall conform to the telemetry source packet structure described in this document (i.e. no segmentation of big source packets is allowed).

5.1.1.3.2 Source Sequence Count

Used to represent the actual Sequence Count. A separate source sequence count is maintained for each Application Process ID and shall be incremented by 1 whenever the source (APID) releases a packet. Therefore the counter corresponds to the order of release of packets by the source and enables the ground to detect missing packets. Ideally, this counter should never re-initialise, however under no circumstances shall it "short-cycle" (i.e. have a discontinuity other than to a value zero). The counter wraps around from 214-1 to zero, and shall start at zero at power on of the unit or on start of the application generating the packet data.

5.1.1.4 Packet Data Length

The Packet Data Length field specifies the number of bytes contained within the Packet Data Field. The number is an unsigned integer "C" where :

C = (Number of bytes in Packet Data Field) - 1

For ASTROSAT-250, the maximum length of a Telemetry Source Packet Data Field is 4090 bytes for OBC generated TM packets, which means Cmax = 4089, including packet secondary header, source data and Packet Error Control.

5.1.2 Packet Data Field

5.1.2.1 Packet Secondary (Data Field) Header

The packet secondary header (or data field header) precedes by the source data in the telemetry packet, refer to Figure 5.1 1. All data field headers have the same basic structure, as follows:

Spare	PUS Version	Spare	Service Type	Service SubType	Destination ID	S/C Time
1 bit	3 bits	4 bit	8 bits	8 bits	8 bits	48 bits
Fixed bit string	Enumerated	Fixed bit string	Enumerated	Enumerated	Enumerated	Enumerated

Figure 5.1-B : Data Field Header

5.1.2.1.1 Spare

All spares shall be set to all zeros, i.e. '0000'BIN.

5.1.2.1.2 PUS Version

Set to value one, e.g. '001'BIN.

5.1.2.1.3 Service Type

This indicates the type to which the telemetry source packet relates. The telemetry source packet types applicable to ASTROSAT-250 are as given in Appendix 1. The numbering scheme is detailed in section A1.3.

5.1.2.1.4 Service Sub-type

Together with the Type, the Sub-type uniquely identifies the nature of the telemetry contained within the telemetry source packet.

Note: The location has to be clarified and agreed early in the system level design phase.

5.1.2.1.5 Destination ID

This field identifies the destination of the telemetry source packet: For AS250, the unique destination for TM packets is the ground but the Destination ID is used as follows:

- For TM(1,X), the destination ID mirrors the telecommand Source ID field of the packet data field header.
- For other telemetry the field shall be set to all zeros, i.e. '0000'BIN = Ground

5.1.2.1.6 S/C Time

This defines the time that the acquisition of the data in the packet was initiated, or another time such that the acquisition time can be calculated by the ground segment using data supplied in the ASTROSAT-250 Users Manual.

The time code format shall be the CCSDS Unsegmented Code (CUC), PTC = 9 and PFC = 17, as defined in § 4.3.3.7.

The S/C Time epoch is 01/01/2000. It is based on the OBC SpaceCraft Elapsed Time (SCET) counter, initialized at 0 on OBC power ON and can be adjusted on ground command to an absolute value (GPS or UTC time).

5.1.2.2 Source Data

The packet source data constitutes the data element of the telemetry reports to the ground. It is made of an integer number of bytes.

5.1.2.3 Packet Error Control Field

The last field, Packet Error Control (PEC, 16bits), shall transport an error detection code (checksum) that can be used by the ground to verify the integrity of the complete telemetry source packet.

The checksum shall be calculated over the complete packet less the final 16 bits Packet Error Control field. Appendix 3 provides a specification of the checksum method selected (CRC checksum).

5.2 TELEMETRY TRANSFER FRAME

All telemetry Transfer Frames must conform to the structure defined in [AD-01].

TM TRANSFER FRAME

TRANSFER FRAME PRIMARY HEADER	TRANSFER FRAME SECONDARY HEADER	TRANSFER FRAME DATA FIELD	TRANSFER FRAME TRAILER	
			OPERATIONAL CONTROL FIELD	FRAME ERROR CONTROL FIELD
6 Bytes	4 Bytes	Varies	4 Bytes	0 Bytes

The Frame Error Control Field has been suppressed : on AS250, the Reed-Solomon code will always be used and, in this case, the Frame Error Control Field is not required anymore. Refer to [AD-01].

TRANSFER FRAME PRIMARY HEADER (6 bytes)

MASTER CHANNEL ID		VIRTUAL CHANNEL ID	OPERATIONAL CONTROL FIELD FLAG	MASTER CHANNEL FRAME COUNT	VIRTUAL CHANNEL FRAME COUNT	TRANSFER FRAME DATA FIELD STATUS
TRANSFER FRAME VERSION NUMBER	SPACECRAFT ID					
2 bits	10 bits	3 bits	1 bit			
2 Bytes				1 Byte	1 Byte	2 Bytes

TRANSFER FRAME DATA FIELD STATUS (2 bytes)

TRANSFER FRAME SECONDARY HEADER FLAG	DATA FIELD SYNCHRONISATION FLAG	PACKET ORDER FLAG	SEGMENT LENGTH ID	FIRST HEADER POINTER
1 Bit	1 Bit	1 Bit	2 Bits	11 Bits

TRANSFER FRAME SECONDARY HEADER (4 bytes)

TRANSFER FRAME SECONDARY HEADER ID		TRANSFER FRAME SECONDARY HEADER DATA FIELD
TRANSFER FRAME SECONDARY HEADER VERSION NUMBER	TRANSFER FRAME SECONDARY HEADER LENGTH	
2 Bits	6 Bits	
1 Byte		3 Bytes

Figure 5.2-A : Telemetry Transfer Frame

5.2.1 Transfer Frame Length

The only allowed frame length (before encoding) shall be 1115 bytes (i.e. 8920 bits).

5.2.2 Version Number

The Version Number shall be set to 00BIN.

5.2.3 Spacecraft ID

The Spacecraft Identification will be assigned for each AS250 spacecraft in a specific space to ground ICD.

5.2.4 Virtual Channel ID

Data sources on board will be allocated a Virtual Channel (VC) number to identify them to the ground processing facilities. The VC allocation will assigned for each AS-250 mission with the following constraints:

- VC0 is assigned to real-time telemetry
- VC1 to VC5 can be assigned for downlink of OBC mass memory packet stores
- VC6 is reserved for OBC housekeeping module
- VC7 is reserved for Idle frames

The term «Idle Frame» means a Transfer Frame containing (only) Idle Data in its Transfer Frame Data Field. In fact, a Transfer Frame containing (only) Idle Data in its Transfer Frame Data Field, can still carry non-idle information outside the Transfer Frame Data Field and then be used for «active» purposes (e.g. extraction of the CLCW in the OCF for the Telecommand protocol; reference Time for time calibration procedures, etc.).

Other Virtual Channels may contain Idle Packets if no data is available when they have to be transmitted.

The priority scheme for downlinking the VCs will have to be defined for each AS-250 mission. This default priority sheme will be contained in the CSW EEPROM image and applied at SW initialisation.If necessary, the ground will be able to modify in-flight the VC priority scheme by using Service 6 on BAT registers.

An example of a default VC priority scheme is as follows :

1. VC0 when ready or when a time packet should be sent
2. VC4 when ready and no VC0 available
3. VC2 when ready and no VC0 or VC4 available
4. VC1 when ready and no VC0 or VC4 or VC2 available

5. VC3 when ready and no VC0 or VC4 or VC2 or VC1 available

6. VC7 Idle Frames - when no other data is available

Note: If a VC frame is not completely filled with real data, the VC frame shall be filled up with one (or more) idle packet(s) and sent before VC7 begins to send idle frames.

5.2.5 Operational Control Field Flag

The Operational Control Field Flag shall be set to 1 and a Command Link Control Word (CLCW) shall be inserted in the Operational Control Field (OCF) for all frames.

5.2.6 Master Channel Frame Count Field

The Master Channel Frame Count field shall contain a sequential binary count (modulo 256) of each Transfer Frame transmitted within the ASTROSAT-250 specific Master Channel. A re-setting of the MASTER CHANNEL FRAME COUNT before reaching 255 shall not take place unless it is unavoidable. Any case when it is unavoidable shall be documented in the Spacecraft user manual.

5.2.7 Virtual Channel Frame Count Field

The Virtual Channel Frame Count field shall contain a sequential binary count (modulo 256) of each Transfer Frame transmitted through a specific Virtual Channel of a Master Channel. A re-setting of the Virtual Channel Frame Count before reaching 255 shall not take place unless it is unavoidable. Any case when it is unavoidable shall be documented in the Spacecraft user manual.

5.2.8 Secondary Header Flag

The Secondary Header shall always be set to one indicating a secondary header shall be inserted in the frame.

5.2.9 Data Field Synchronisation Flag

The Data Field Synchronisation Flag shall be set to zero; i.e. byte-synchronised and forward-ordered Telemetry Source Packet or Idle Data (only for VC7) shall be inserted in the Transfer Frame Data Field.

5.2.10 Packet Order Flag

The Packet Order Flag shall be set to zero. The Packet sequence count order shall be forward.

5.2.11 Segment Length Identifier

Since the Data Field Synchronisation Flag is set to zero, the Segment Length Identifier shall be set to 11BIN.

5.2.12 First Header Pointer

Since the Synchronisation Flag is set to zero, the First Header Pointer shall contain information on the position of the first Telemetry Source Packet within the Transfer Frame Data Field; i.e. the binary representation of the location of the first byte of the first Packet Primary Header. The locations of any subsequent headers within the same Transfer Frame Data Field will be determined by calculating these locations using the Packet Data Length Field.

If no Packet Primary Header starts in the Transfer Frame Data Field, the First Header Pointer shall be set to «11111111111» BIN.

For Idle Frames (VC7) the First Header Pointer shall be set to «11111111110» BIN.

5.2.13 Secondary Header

A Transfer Frame Secondary Header shall be inserted in all frames. This shall contain a header and an expansion of the virtual channel frame counter.

5.2.13.1 Secondary Header ID

The secondary header ID shall be 8 bits in length and shall indicate the version number and the header length for ASTROSAT-250, this shall be set to 00000011 BIN.

5.2.13.2 Secondary Header Data

The secondary header data shall be a 3 Byte field containing an additional 24 bits of the virtual channel frame count as defined in [AD-01].

5.2.14 Operational Control Field

The Operational Control Field shall be inserted in each frame and it shall contain the CLCW (i.e. the Control Word Type shall be set to zero) with the format defined in [AD-02].

CONTROL WORD TYPE "0"	CLCW VERSION "00"	STATUS FIELD	COP IN EFFECT	VIRTUAL CHANNEL IDENTIFICATION	SPARE
1	2	3	2	6	2

(ALWAYS "0" FOR CLCW)

FLAGS					FARM B COUNTER	SPARE	REPORT VALUE
NO RF AVAIL	NO BIT LOCK	LOCK-OUT	WAIT	RETRANSMIT			
1	1	1	1	1	2	1	8

Figure 5.2-B : Operational Control Field

The CLCW is inserted in the TM frame alternatively from each TC decoder: CLCWs with a VC ID belonging to the nominal Command Decoder are transmitted in the Transfer Frames with an even Master Channel Frame Count and CLCWs with a VC ID belonging to the redundant Command Decoder are transmitted in the Transfer Frames with an odd Master Channel Frame Count.

5.2.14.1 Frame Error Control Word

The Frame Error Control Word is not used.

5.2.14.2 Bitstream Pseudo Randomiser

An on-board pseudo randomiser shall be implemented. The requirement for the pseudo randomiser are given in [AD-04].

6 TELECOMMAND STRUCTURE

6.1 TELECOMMAND SOURCE PACKETS

All telecommand source packets must conform to the structure defined in [AD-03] and tailored for AS250 in Figure 6.1 1 below.

<table>
<tr><th colspan="8">PACKET PRIMARY HEADER (48 bits)</th><th colspan="3">PACKET DATA FIELD (VARIABLE)</th></tr>
<tr><td colspan="5">PACKET ID</td><td colspan="2">PACKET SEQUENCE CONTROL</td><td rowspan="5">PACKET LENGTH</td><td rowspan="5">SECONDARY HEADER (OPTIONAL)</td><td rowspan="5">APPLICATION DATA</td><td rowspan="5">PACKET ERROR CONTROL</td></tr>
<tr><td rowspan="3">VERSION NUMBER</td><td rowspan="3">TYPE</td><td rowspan="3">SEC. HDR. FLAG</td><td colspan="2">APPLIC. PROCESS ID</td><td rowspan="3">SEQUENCE FLAGS</td><td rowspan="3">SEQUENCE COUNT</td></tr>
<tr><td>PID</td><td>PCAT</td></tr>
<tr><td>7</td><td>4</td></tr>
<tr><td>3</td><td>1</td><td>1</td><td colspan="2">11</td><td>2</td><td>14</td></tr>
<tr><td colspan="5">2 Bytes</td><td colspan="2">2 Bytes</td><td>2 Bytes</td><td>4 Bytes</td><td>(VARIABLE)</td><td>2 bytes</td></tr>
</table>

Up to 1018 bytes

Figure 6.1-A : Telecommand Packet Fields

For AS250, the maximum size of a Telecommand Source Packet (TC packet) has been fixed to 1024 bytes. As a result, the maximum size of the Packet Data Field is 1018 bytes.

6.1.1 Packet Primary header

6.1.1.1 Packet Identification

6.1.1.1.1 Version Number

The Version Number must be set to '000' BIN for all commands.

6.1.1.1.2 Packet Type

This bit distinguishes between telecommand packets and telemetry source packets. For telecommand packets, the type = 1.

6.1.1.1.3 Secondary Header Flag

This indicates the presence of the Data Field Header when set to 1.

For ASTROSAT-250 all commands (except CPDU commands described in §6.1.3) will have a data field header.

6.1.1.1.4 Application Process ID

The application ID (APID) is structured into two fields :

- The Process ID (PID) which defines the application which the telecommand is addressed to.
- The Packet category (PCAT) is fixed to decimal 12.

The choice of Process ID values across the spacecraft subsystems and experiments are given in Appendix 2.

6.1.1.2 Packet Sequence Control

6.1.1.2.1 Sequence Flags

For the TC packets, the Sequence Flags are fixed to value 11 BIN meaning “stand-alone” packet.

6.1.1.2.2 Sequence Count

The Sequence Count (14 bits) is maintained by the telecommand source for each telecommand source and each Application Process ID. The sequence count shall be incremented by 1 whenever a command is generated with that Application Process ID. The counter wraps around from "full-scale" to zero.

When an acknowledgement of a packet is required (see "Ack" field in the data field header below), it is mandatory that the Sequence Control field is included in the telemetry acknowledge packet as the identifier of the telecommand packet being acknowledged (see 7.1).

No check is to be performed by the addressed application regarding sequence counter, the application should accept commands regardless of the sequence counter.

6.1.1.3 Packet Length

The Packet Length field specifies the number of bytes contained within the Packet Data Field. The number is an unsigned integer "C" where

C = (Number of bytes in Packet Data Field) - 1

Maximum length of a Telecommand source packet data field is 1018 bytes (C = 1017). This includes 4 bytes data field header, 1012 application data and 2 packet error control.

Since the total maximum size of a TC is limited to 1024 bytes, the TC length (including header and PEC) for commands to be inserted into the MTL thanks to TC(11,4) can not be larger than 1024 – packet header – 2ndary header – Sub Schedule ID - N – TT - PEC = 1024 - 6 – 4 - 1 -1 - 6 - 2 = 1004 bytes (considering only one command inserted in a TC(11,4), i.e. N = 1). This results in a maximum packet length (C) for time tagged commands of 1004 - 6 - 1 = 997 bytes.

Following table summarizes the Packet length field maximum value for ASTROSAT-250 telecommands:

Telecommand	Data field max size	Packet max length (C)
TC(11,4): Insert telecommands in the command schedule	1018	C_{max} = 1017
Any TC as parameter of TC(11,4)	998	C_{max} = 997
Any TC as parameter of TC(19,1) : Add an event to detection list	1003	C_{max} = 1002
Any TC as parameter of TC(19,1) inserted into TC(11,4)	983	C_{max} = 982

Table 6.1-B : Packet length field maximum value

6.1.2 Packet Data Field of non CPDU commands

6.1.2.1 Data field Secondary Header

The data field secondary header is preceded by the packet header and followed by application data and error control in the telecommand packet, refer to Figure 6.1 1. The data field secondary header is defined as follows:

CCSDS Secondary header Flag	PUS Version	Ack	Service Type	Service Subtype	Source ID
Boolean	Enumerated	Enumerated	Enumerated	Enumerated	Enumerated
1 bit	3 bits	4 bits	8 bits	8 bits	8 bits
4 bytes					

Figure 6.1-C : Data field secondary header

6.1.2.1.1 CCSDS Secondary header flag

Shall be set to zero to indicate that this header is a non-CCSDS defined header.

6.1.2.1.2 PUS Version

It shall be set to 1.

6.1.2.1.3 Ack

This field indicates the acknowledgements required in the form of telemetry packets to verify acceptance and execution of this telecommand packet.

The bit settings defined for ASTROSAT-250 are as follows (with bit zero as start of the data field header):

- ---x = Acceptance of packet by application: this field shall be set to 1 to indicate that an acknowledge report is required.
- --x- = 0 Not used (Acknowledge start of execution)
- -x-- = 0 Not used (Acknowledge progress of execution)
- x--- = Acknowledge completion of execution: this field shall be set to 1 to indicate that an acknowledge report is required.

All applications, which receive telecommands, must generate acknowledgements as specified in the telecommand message.

An encapsulated Telecommand packet shall be acknowledged separately from its transport command depending on the Ack flag of each command.

6.1.2.1.4 Service Type

This identifies the service-type to which the telecommand packet relates. The telecommand source packet types applicable to ASTROSAT-250 are given in the Appendix 1.

6.1.2.1.5 Service Subtype

Together with the Type, the Sub-type uniquely identifies the nature of the command contained within the telecommand packet.

The relationship between Telecommand and Telemetry Packet Types and Subtypes and the users to which they apply is given in §7.

The same Packet Type and Subtype definitions shall apply to all applications.

6.1.2.1.6 Source ID

This field indicates the source of the command, ground or on-board process, to identify a particular telecommand packet so that it can be traced within the end-to-end telecommand system. Its value is set by the ground and is mirrored in the Destination ID field of the telecommand acknowledges TM(1,X).

The ground has the flexibility to use different source values or not. For instance, following values can be used to distinguish commands sent directly from the ones uploaded in PUS services 11, 18 or 19.

Source ID (hex)	TC Source
0x00 (default)	Ground
0x11	PUS service 11 (MTL)
0x18	PUS service 18 (OBCP)
0x19	PUS service 19 (event/action)

Table 6.1-D : TC source ID allocation example

6.1.2.2 Spare

Spare bits shall be introduced in order to pad the data field to an integral number of words (byte or longer). All spare bits shall be set to zero.

6.1.2.3 Application Data

The telecommand application data constitutes the data element of the command.

6.1.2.4 Packet Error Control

The last field, Packet Error Control (PEC, 16bits), provides an error detection code (checksum) in the packet allowing the receiving application to verify the integrity of the telecommand packet data.

The checksum shall be calculated over the complete packet less the final 16 bits Packet Error Control field.

Appendix 3 provides a specification of the checksum method selected (CRC checksum).

6.1.3 CPDU telecommand (DTC) packets

Each Reconfiguration Unit (RU A & B) is able to issue CPDU commands (also called DTC), for driving vital spacecraft functions. On AS250, CPDU commands offer several services :

- Generation of High Power Commands (HPC)
- Set PM master (A or B)
- Generation of OBC static configuration commands :
  - Select TFG (A or B) on each OBC (A or B)
  - Enable / disable authentication on each TC Decoder (A or B)
  - Lock / unlock authentication on each TC Decoder (A or B)
  - Reset PM
  - Reset OBC
  - Etc

All CPDU commands services are detailed in [RD-03].

All CPDU commands are processed directly by the TC decoder (MAP-ID = 0 in the TC segment as described in 6.2) and the CSW is never involved.

A CPDU command packet (or DTC packet) has a primary header as defined in §6.1.1 and a packet error control field as defined in §6.1.2.4 but no secondary header (secondary header flag shall be set to '0'). It can not be segmented (Packet sequence flags shall be '11') so it shall be carried inside a single telecommand segment.

The Application process identifier shall be PID= "OBC hardware" and PCAT= "TELECOMMAND".

Each CPDU command packet carries command instructions and have the following packet data field structure:

CPDU PACKET DATA FIELD					
N command Instruction(s)					PACKET ERROR CONTROL
Command Instruction 1	Command Instruction 2	Command Instruction 3	…	Command Instruction N	
16 bits	16 bits	16 bits		16 bits	
2*N bytes					2 bytes

Figure 6.1-E : CPDU commands Packet Data Field

The CPDU command application data is made of N elementary command instructions ($N \leq 115$) and a Packet Error Control Field. The structure of the elementary CPDU command instruction is detailed in [RD-03].

Packet Error control field is generated over the entire CPDU command Packet data field (except the Packet Error Control filed itself) using CRC checksum as specified in [RD-04].

The processing of CPDU commands is detailed in [RD-04].

6.2 TELECOMMAND SEGMENTS

The Telecommand Segment defined in [AD-02], and shown in the figure below, shall be used as TC Frame Data Unit (i.e. the data unit transferred from the Segmentation Layer to the Transfer layer to be inserted in the Frame Data Field of the Telecommand Frame).

The Segment Header contains the following two fields:

- Sequence Flags (Bits 0,1), and
- Multiplexer Access Point (MAP) Identifier (Bits 2 through 7)

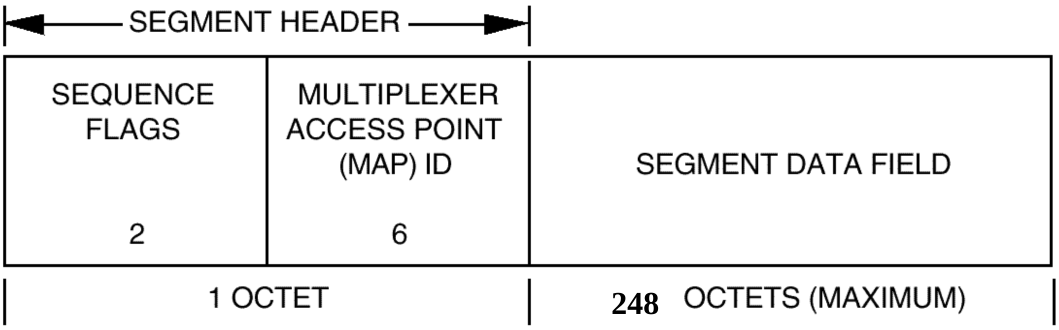
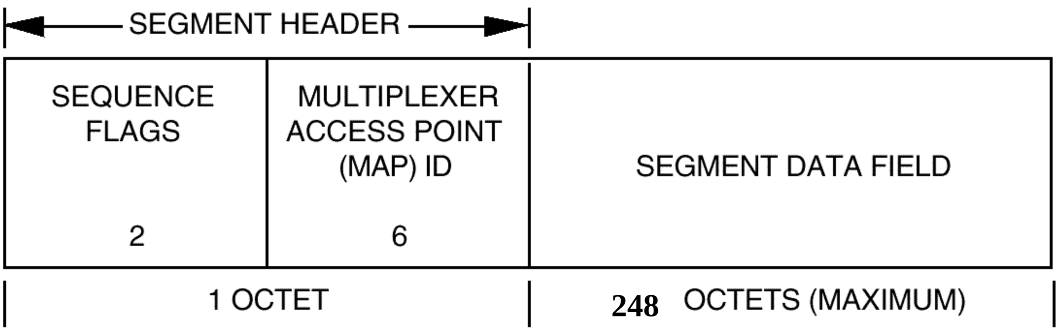


Figure 6.2-A : Telecommand Segment

6.2.1 Sequence Flags

AS250 CSW is compatible with segmentation to allow to receive TC packets with a size bigger than 248 bytes. In order to transmit a TC packet of the maximum size (1024 bytes for AS250), the ground will use 5 segments Therefore, AS250 DMS software will be sized to accept a maximum of 5 segments. If the ground cannot manage segmentation, then it will be limited to send TC packets of 248 bytes maximum.

The 2 bit field delimits the higher layer TC User Data Unit by indicating the sequential position of the TC Segment relative to the TC User Data Unit (e.g. packet) of which the TC Segment is a part. The flags are interpreted as follows:



6.2.2 Multiplexer Access Point (MAP) Identifier

MAP ID's shall be used to route the telecommands from the decoder depending on the type of handling required for the command e.g. OBC software or Command Pulse Distribution Unit. MAP ID's shall not be used to address the currently active DMS processor.

MAP-ID = "CPDU" = 0 is used for CPDU commands.

MAP-ID = "Normal" = 1 is used for normal commanding other than CPDU command

MAP-ID = "MAP 63" = 111111 is used for Authentication Control Commands

6.2.3 Packet Aggregation

As seen above, the Segment Data Field can contain either a portion of a TC Packet, or a complete TC Packet or an aggregation of TC Packets (multiple complete packets).

In order to maximise the throughput of commands on the uplink, packet aggregation may be used where possible, by the ground. Aggregation is a CCSDS concept where several complete packets can be put into a single segment. Therefore at the start of a segment there will always be the start of a packet, the length of the first packet will define the start position of the next packet. The aggregation is not mandatory for the ground and the DMS will manage all packets within the Segment Data Field, aggregated or not.

Segment Data Field		
Packet #1	Packet #2	Packet #3

(« Packet Length » of packet #1)+7bytes = Start address Byte of Packet #2

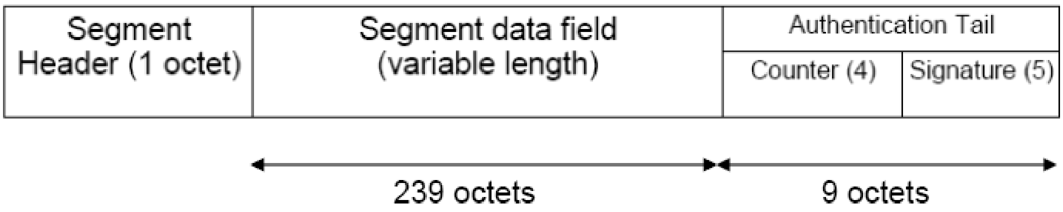
Start address Byte of Packet #2 + (« Packet Length » of packet #2) + 7bytes =

Start address Byte of Packet #3

Important note : there shall be no aggregation for CPDU commands. In other words, there shall be only one packet of CPDU commands (DTC) in the Segment Data Field of a CPDU Segment.

6.2.4 Authentication

AS250 OBC offers a security mechanism for telecommands : the Authentication. Data authentication is achieved by identifying an Authentication Tail at the end of the Telecommand Segment, which content is generated by a cryptographic algorithm. This extra unit of information is a Digital Signature which identifies definitely the origin of the data. More details can be found in [RD-01]. The use of the Authentication will automatically reduce the size of the Segment Data Field. This is entirely managed by the OBC SCOC3.



Authentication Control Commands, identified by MAP-ID 63, allow to configure the authentication : select fixed or programmable key, change fixed or programmable key, set new LAC Counter. This is detailed in [RD-02]. Authentication Control Commands are entirely managed by the OBC SCOC3 and they do not support segmentation.

CPDU commands (DTC) are used to activate / deactivate and lock / unlock the Authentication mode. This is detailed in [RD-03].

6.3 TELECOMMAND FRAME

The Telecommand function shall support the COP-1 procedure in accordance with [AD-02].

Within COP-1, the packet telecommand services AD (Sequence Controlled Service) and BD (Expedited Service) shall be supported in parallel.

The Telecommand Frame must conform to the structure defined in [AD-02].

TC TRANSFER FRAME

TRANSFER FRAME PRIMARY HEADER	TRANSFER FRAME DATA FIELD	FRAME ERROR CONTROL FIELD
5 Bytes	Variable	2 Bytes
	Up to 251 bytes	

TRANSFER FRAME PRIMARY HEADER

TRANSFER FRAME VERSION NUMBER	BYPASS FLAG	CONTROL COMMAND FLAG	SPARE	SPACECRAFT ID	VIRTUAL CHANNEL ID	FRAME LENGTH	FRAME SEQUENCE NUMBER
2 bits	1 bits	1 bits	2 bits	10 bits	6 bits	10 bits	8 bits
2 Bytes					2 Bytes		1 Byte

Figure 6.3-A : Telecommand Transfer Frame

The Telecommand Transfer Frames are of 3 types (AD, BD and BC) as indicated by the Bypass Flag and the Control Command Flag. The definition and the use of these flags are detailed in [AD-02].

6.3.1 Spacecraft ID

Refer to § 5.2.3.

6.3.2 Virtual Channel ID

The following two Virtual Channels Identifiers shall be used addressing the two on-board decoders (all TC will have the same behaviour on both TC decoders):

VC ID	TC Decoder
0x1	A
0x2	B

These two values shall be used consistently in the CLCWs.

On AS250, all Telecommand Transfer Frames received on-board are routed to both TC Decoder (A and B). But, a Telecommand Transfer Frame will be processed by only one TC Decoder : the one addressed by the VC ID.

6.3.3 Frame Length

This 10-bit field contains a length count «C» which equals one fewer than the total bytes in the TC Transfer Frame. The length count «C» is expressed as:

«C» = (Total Number of Bytes) - 1

The count is measured from the first bit of the FRAME HEADER to the last bit of the FRAME ERROR CONTROL FIELD. The size of this field limits the maximum length of a TC Transfer Frame to 1024 bytes, however, the maximum frame length applicable to ASTROSAT-250 is 256 bytes. Therefore the first two bits of the Frame Length field must always be set to 00BIN, leaving an effective "Frame Length Field" of 8 bits.

6.4 TELECOMMAND LOWER LAYERS

All the telecommand lower layers data structures and procedures must conform to the structure defined in [AD-02].

7 TM AND TC PACKET TYPES AND SUBTYPES

The ESA Packet Utilisation Standard (PUS) has been used to generate the standard services together with other packet services identified from the ASTROSAT-250 top level design.

Following table shows the Packet Service Types applicable to the AS-250 CSW.

The legend to the table is as follows:

- A = Applicable
- N/A = Not Applicable
- NU = Not used

Service Type	Service Name	AS-250 CSW Applicability
1	Telecommand Verification Service	A
2	Device Commanding Service	A
3	Housekeeping and Diagnostic Data Reporting Service	A
4	Parameter statistics Reporting Service	NU
5	Event Reporting Service	A
6	Memory Management Service	A
7	Not defined	N/A
8	Function Management Service	NU(1)
9	Time Management Service	A
10	Not defined	N/A
11	On-board Mission Time Line Service	A
12	On-board Monitoring Service	A
13	Large Data Transfer Service	NU
14	Packet Forwarding Control Service	A
15	On-board Storage and Retrieval Service	A
16	Not defined	N/A
17	Test Service	A
18	On-Board Operations Procedure Service	A
19	Event/Action Service	A

Table 7-A : Standard services applicable to ASTROSAT-250

(1) Each OBC software function implementing mission specific needs may be operated through private telecommand. So for ASTROSAT-250 it is proposed not to implement PUS service #8 and just create the private telecommands necessary to operate on-board software functions.

NOTES :

In order to avoid duplication of information, this document defines the selection of the AS250 TM and TC services but the detailed structures are provided by the SW TM/TC ICD.

7.1 SERVICE 1 : TELECOMMAND VERIFICATION

Objective

This service allows the command source to verify identified commands at acceptance and/or execution by asking the addressed application to generate service type 1 reports in the telemetry stream.

Description

The command source can set two bits in the command packet header, one asks for an acceptance report; the other an execution report, the two bits can be set to any value. CSW uses these bits to generate the required reports; No systematic on-board check is done on the report this may be done by the command sender if required. An acceptance report is generated immediately after completion of checks on validity of the packet header; an execution report after internal verification of TC execution.

Service 1 is used to verify that the addressed application has received and executed the packet telecommand addressed to it. The type of response required is coded in the Acknowledge field contained in the telecommand packet header. For ASTROSAT-250, the response required is restricted to:

- No Response (acceptance / execution success acknowledge report not required, not applicable to acceptance and execution failure)
- Acceptance Success or Failure (service report sub-type 1 or 2 required)
- Execution Success or Failure (service report sub type 7 or 8 required)

The type of response required for each command depends on the function of the command and is coded with the command definition in the Spacecraft Data Base.

Notes

Each TC packet received is submitted to the checks defined here below independently from the ACK flags settings, and after the checks on the TC segments (i.e. checks of segmentation errors and checks on segments received bytes w.r.t. the packet length).

Static Acceptance Checks (eventually issuing a Telecommand Acceptance Failure Report):

- Check the indicated TC packet length w.r.t. the packet structure defined in § 6.1
- Compute packet error control word and check w.r.t. received packet error control word
- Check the constant fields in the packet header (version number, type, data field header flag, and sequence flag) and data field header (PUS version)
- Check the APID:
  - Check the PID w.r.t. the assigned PID number(s)
  - Check the field PCAT (always 12 for TC)

- Check whether Service Type/Subtype is supported by the destination PID

Consistency Checks (eventually issuing a Telecommand Execution Completion Failure Report):

- For TC with parameters, check the actual TC length w.r.t. expected TC length associated with actual service type and service subtype.
- Check whether parameters included in the Source Data Field are within their defined range (specific for a Service type/subtype).

Notes: TC consistency checks shall only be performed after all static checks have been passed successfully. TC execution shall only start after all consistency checks have been passed successfully.

In addition to the consistency checks execution success checks (specific for a Service type/subtype, e.g. read back written data from H/W) may be performed, before eventually a Telecommand Execution Completion Report is issued.

In the case of failures, the command acceptance / execution failure reports will be generated whatever the value of the corresponding bit in the secondary data field header of the TC packet.

Sub-services

Notes: TC consistency checks shall only be performed after all static checks have been passed successfully. TC execution shall only start after all consistency checks have been passed successfully.

In addition to the consistency checks execution success checks (specific for a Service type/subtype, e.g. read back written data from H/W) may be performed, before eventually a Telecommand Execution Completion Report is issued.

Table 7.1-A : Service 1 sub-services

7.2 SERVICE 2 : DEVICE COMMAND DISTRIBUTION

Objective

The device command distribution service provides the capability for the distribution of 1553 bus command messages.

On some projects, this Service often includes some other hardware services. They are not foreseen on AS250 for the following reasons :

- For OBC HPC commanding, specific services will be created to handle these commands consistently with the related SW modes and functions.
- Read and Load registers of the OBC or of the Reconfiguration Unit (RU). These services are provided by Service 6. With the AS250 OBC, the access to registers is identical to the access to memory.

1553 MIL-STD commands

1553 bus command messages can be the following:

- Bus Controller (BC) to Remote Terminal (RT) transfer, so called 'receive command' as the remote terminal is to receive data
- RT to BC transfer, so called 'transmit command' as the remote terminal is to transmit data.
- Mode command with the following subcategories:
  - Transmit without data
  - Transmit with data
  - Receive with data

Transfers are under the responsibility of the BC only. The BC can access to both channels of each bus by using dedicated bus coupler.

The service 2 provides the capability to group two 1553 bus command messages into a single TC(2,128). The central software extracts each command message from the TC and sends them on the selected bus. The number of TC(2,128) that can executed per second and the delay between the 2 command messages will be defined for each AS250 mission and it will then be constant for the whole duration fo the mission. Message answers are then downlinked thanks to Service 2 TM (one TM per command message).

Two different 1553 buses are considered on AS250 :

- the service module bus or Avionic Bus (AVB)
- the payload module bus (PLB)

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(2,128)	Send 1553 command messages	DMS
TM(2,129)	1553 bus command message answer	DMS

Table 7.2-A : Service 2 sub-services

7.3 SERVICE 3 : HOUSEKEEPING AND DIAGNOSTIC REPORTING

Objective

This service controls the generation of CSW periodic report packets, including data from non-packet based users. This service, along with the event reporting service, provides for the reporting to the ground, all information of operational significance that is not explicitly provided within the reports of another service. The service consists of two independent sub-services which cover, respectively, the requirements for:

- housekeeping data reporting;
- diagnostic data reporting.

Any number of on-board application processes may provide a single instance of the housekeeping and diagnostic data reporting service.

Description

Generation start, stop, frequency and content of housekeeping report packets are controlled by this service.

Housekeeping

The housekeeping data reporting sub-service samples sets of housekeeping parameters in accordance with a set of reporting definitions stored onboard. There will be a pre-defined set of such definitions onboard as deemed appropriate for the housekeeping monitoring of the mission. However, these definitions may be modified, deleted and new definitions may be added by the ground at any time.

A Structure Identification (SID) is associated with each distinct reporting definition and associated housekeeping report packet. The SID will be used on the ground, together with the Application Process ID and knowledge of the nature of the packet (i.e. that it is a housekeeping packet, as opposed to a diagnostic packet), Service Type and Sub-type to identify the housekeeping report packet and to interpret its content. The SID shall be unique to a given service implementation and packet nature (i.e. housekeeping or diagnostic), however different instances of the service within different application processes can use the same values of SID.

Diagnostic service concept

The diagnostic data reporting sub-service shall be functionally identical to the housekeeping data reporting sub-service. Different service subtypes shall be used, however, primarily for the purposes of distinguishing the diagnostic parameter reports for routing and (ground) processing.

A means to disable the generation of certain diagnostic parameter reports (whose definitions can remain on-board for intermittent use, for example, when a particular anomaly occurs) shall be provided. Because of the nature of diagnostic mode, it is anticipated that the parameter reports contain a predominance of fixed-length arrays corresponding to parameters sampled at very high rates, many times per report.

SID's allocated to Housekeeping and Diagnostic data will be defined for each AS250 mission according to the following ranges:

SID Range	Allocated to
0	Reserved
1 to 10	Housekeeping in Real-time and Playback (HK_RT category)
11 to 20	Housekeeping in Playback only (HK_PB category)
21 to 30	Diagnostic (DIAGNOSTIC category)

Table 7.3-A : SID allocations

Data Collection

Each reporting definition has an associated data collection interval, which is the time interval over which the housekeeping parameters are sampled. Parameters within a reporting definition are sampled only once per collection interval.

In addition, the data sampling, data collection and parameter-report-generation activities for a housekeeping parameter report may be temporarily disabled (e.g. to reduce the on-board processing load).

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(3,1)	Define new HK parameter reports	All PIDs
TC(3,2)	Define new diagnostic parameter reports	All PIDs
TC(3,3)	Clear HK parameter report definitions	All PIDs
TC(3,4)	Clear diagnostic parameter report definitions	All PIDs
TC(3,5)	Enable HK parameter report generation	All PIDs
TC(3,6)	Disable HK parameter report generation	All PIDs
TC(3,7)	Enable diagnostic parameter report generation	All PIDs
TC(3,8)	Disable diagnostic parameter report generation	All PIDs
TC(3,9)	Report HK parameter report definitions	All PIDs
TM(3,10)	HK parameter report definitions report	All PIDs
TC(3,11)	Report diagnostic parameter report definitions	All PIDs
TM(3,12)	Diagnostic parameter report definitions report	All PIDs
TM(3,25)	Housekeeping parameter report	All PIDs
TM(3,26)	Diagnostic parameter report	All PIDs
TC(3,128)	Request HK parameter report	All PIDs
TC(3,129)	Define HK parameter report Collection Interval	All PIDs
TC(3,130)	Define Diagnostic parameter report collection interval	All PIDs
TC(3,132)	Append parameters to existing HK definition	All PIDs
TC(3,133)	Append parameters to existing Diagnostic definition	All PIDs

Table 7.3-B : Service 3 sub-services

7.4 SERVICE 4 : PARAMETER STATISTICS (NOT USED)

Service not included in current AS-250 baseline.

7.5 SERVICE 5 : EVENT REPORTING

Objective

This service provides for the reporting to the service user of information of operational significance which is not explicitly provided within the provider-initiated reports of another service. The service covers the requirements for event reporting, i.e.:

a. reporting of failures or anomalies detected on-board;

b. reporting of autonomous on-board actions;

c. reporting of normal progress of operations and activities, e.g. detection of events which are not anomalous (such as payload events), reaching of predefined steps in an operation.

Some reports can combine more than one of these events.

EXAMPLE A report can declare that “Unit X has been switched off because its temperature was detected as 31 ºC, where the currently defined limit is 30 ºC”.

Any number of on-board application processes may provide a single instance of the event reporting service.

Description

This service provides the capability for the generation of reports by any on-board function to notify the ground system of an event of operational significance.

Normal progress of an on-board process is reported via sub type 1. Depending on the process, reporting points will be defined during its design, for key points a time out in the OBC software may be defined.

Anomaly or error reporting is split into three levels, low/Warning (Sub Type 2), medium/Ground action (Sub Type 3) and high/On-board action (Sub Type 4).

The use of the different levels will be defined during the design of the process issuing them. Typically “low” will just be stored for down link to ground, “high” will always have a pre-defined response by the OBC software to recover the anomaly.

Event reports will be one of the prime methods used to control day to day operations during the mission both to report normal progress, warnings, errors requiring ground action or autonomous actions performed on-board.

The generic TM/TC ICD defines commands to enable and disable event report generation as well as one TC to request the a TM report defining the enabled events.

Once generated, events are forwarded toward ground and/or recorded in the Mass Memory according to configuration of service 15.

Event identier (EID)

All EID's reported by the CSW will be defined in the SDB. With the combination of the EID and the PID (the source generating the event) the ground will get from the SDB the complete event description, including the associated parameters providing additional information on the event.

The EID value is assigned per event severity as follows:

- 0x0000 to 0x3FFF: TM(5,1)
- 0x4000 to 0x7FFF: TM(5,2)
- 0x8000 to 0xBFFF: TM(5,3)
- 0xC000 to 0xFFFF: TM(5,4)

Sub-services

Subtype	Sub-service Name	CSW applicability
TM(5,1)	Normal/Progress Report	All PIDs
TM(5,2)	Error/Anomaly Report – Low Severity	All PIDs
TM(5,3)	Error/Anomaly Report – Medium Severity	All PIDs
TM(5,4)	Error/Anomaly Report – High Severity	All PIDs
TC(5,5)	Enable Event Packet Generation	All PIDs
TC(5,6)	Disable Event Packet Generation	All PIDs
TC(5,133)	Report Disabled Event Packets	All PIDs
TM(5,134)	Disabled Event Packets Report	All PIDs

Table 7.5-A : Service 5 sub-services

7.6 SERVICE 6 : MEMORY MANAGEMENT

Objective

This service relates to the management of the various memory areas (e.g. RAM or mass memory unit) which exist on-board the satellite. The service provides the capability for loading, dumping and checking the contents of contiguous memory areas.

Any number of on-board application processes may provide a single instance of the memory management service; however, the number of instances shall ensure that all on-board changeable memory areas can be loaded and that all on-board memory areas can be dumped.

Description

The memory management service provides basic dump, load and check facilities.

A "Memory ID" (MID) uniquely identifies each on-board memory block.

The addressing techniques used on AS250 for memory load, dump and check requests and reports is the absolute addressing. This allows the user to specify a real address start loading or dumping from. The address is expressed in Single Addressable Unit (SAU) corresponding to the one of the selected memory ID. For the DMS, the SAU is fixed to 8 bits.

The memory management service will generate as many TM Dump packets as necessary to cover the entire commanded dump area, such as the TM Dump packets never exceeds the maximum size of a single telemetry source packet.

The service only supports requests on contiguous memory blocks: requests crossing memory boundaries will be rejected. Scatter load feature is not implemented on AS250 : TC(6,2) only contains one block of contiguous memory word(s) to be loaded.

The TM(6,6) Memory dump report is not limited in length. The dump application will generate as many TM dump packets as required to cover the entire commanded dump-area.

In order to avoid having to dump an area of memory after a load the service provides a check command TC(6,9) to check-sum an area (using CRC algorithm as defined in § A.1.1) and downlink the result in a TM(6,10) report.

Memory protections

Some memories are write protected. For HW protected memories, the unprotection/re-protection (throught direct TC to RU) shall be performed by the ground before and after the load memory activation. For the SW protected memory (SGM), the unprotection/ protection is performed by CSW (during the load memory execution).

For EEPROM memories (except SGM), CSW checks the protection status and raise a TM(1,8) if the memory is protected.

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(6,2)	Load memory using absolute addresses	DMS
TC(6,5)	Dump Memory using absolute address	DMS
TM(6,6)	Memory Dump using absolute addresses report	DMS
TC(6,9)	Check Memory using Absolute Addresses	DMS
TM(6,10)	Memory Check using Absolute Addresses Report	DMS
TC(6,130)	Execute PM Jump	DMS

Table 7.6-A : Service 6 sub-services

7.7 SERVICE 7 : NOT DEFINED.

7.8 SERVICE 8 : FUNCTION MANAGEMENT SERVICE (NOT USED).

Service not included in current AS250 baseline.

7.9 SERVICE 9 : TIME MANAGEMENT

Objective

The On-Board Computer (OBC) maintains a satellite time reference called Spacecraft Elapsed Time (SCET). This SCET is compliant to the CUC format (CCSDS Unsegmented Time Code) and has 4 octets of coarse time and 3 octets of fine time. Therefore, the LSB is 2-24 seconds i.e. approximately 59.6 ns. It is managed by the OBC hardware within a register called “OBT counter”.

The satellite On Board Time (OBT) used to timestamp the TM packets and schedule the commands in the Mission Time Line (MTL) is a truncated form of SCET: 4 octets of coarse time and 2 octets of fine time (the LSB is 15.3 µs). The OBT epoch is Agency defined to 01/01/2000 00:00:00.

The SCET (and therefore the OBT) is downlinked periodically to the ground via the “Time Report” TM(9,2) and its generation rate is defined by the ground using TC(9,1). The ground segment will perform the time correlation between OBT and UTC (Universal Time Coordinated) by making use of the Time Source Packets generated on board. The ground is then able to add to the SCET the offset to the wished absolute time reference (e.g. UTC), in order to use this absolute time in the OBT for the space to ground command and control interface.

Description

The OBC clock oscillator used for managing the SCET is enslaved with the GPS 1 Hz pulse signal (PPS) when this one is available. When the GPS is not available, the OBC clock is in free-running mode. When the GPS PPS is available, the DMS starts synchronizing the OBC clock 1 Hz signal with the GPS PPS, using a smooth algorithm. Once the synchronization has been obtained with an accuracy better than 1 µs, the OBC clock is declared “synchronized” and will be maintained synchrone with the GPS as long as it is available.

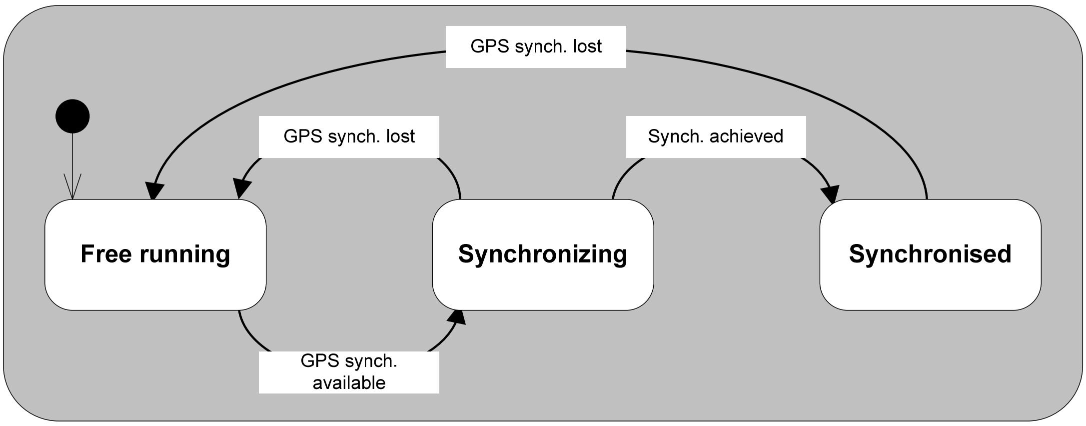


Figure 7.9-A : OBC clock modes

Time reporting

The standard Time Report subservice TM(9,2) provides the capability for the generation of Time Reports, such that the satellite time correlation procedures in the ground segment can be performed. The TM(9,2) packet does not have a data field header.

The rate control sub-service shall maintain the generation rate (one report every 1, 2, 4, 8, 16, 32, 64, 128 or 256 telemetry frames) of the time report. It shall also have means of communicating this generation rate to the time reporting sub-service. When a new generation rate is requested, the time reporting sub-service shall use this new generation rate from the next telemetry transfer frame that meets the above criterion.

Rate	Generation rate
0	$2^0=1$
1	$2^1=2$
2	$2^2=4$
3	$2^3=8$
4	$2^4=16$
5	$2^5=32$
6	$2^6=64$
7	$2^7=128$
8	$2^8=256$

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(9,1)	Change Time Report Generation Rate	DMS
TM(9,2)	Time Report.	DMS
TC(9,128)	Modify SCET	DMS

Table 7.9-B : Service 9 sub-services

7.10 SERVICE 10 : NOT DEFINED

7.11 SERVICE 11 : ON-BOARD OPERATIONS SCHEDULING MANAGEMENT

Objective

The on-board operations scheduling service provides the capability to command on-board application processes using telecommands pre-loaded on-board the satellite and released at their due time. To achieve this, the service maintains an on-board command schedule and ensures the timely execution of telecommands contained therein.

Description

General

The on-board operations scheduling service shall maintain a command schedule which contains telecommand packets and their associated scheduling information.

The service user(s) can request the following activities:

- Enable the scheduling of all, or a subset of, the telecommands in the command schedule (e.g. those to be sent to specified application processes).
- Disable the scheduling of all, or a subset of, the telecommands in the command schedule.
- Add telecommands to the command schedule.
- Delete all, or a subset of, the telecommands in the command schedule (e.g. the telecommands becoming due for release within a specified time period).
- Report on all, or a subset of, the telecommands in the command schedule.
- Report the status of the command schedule.

The command schedule

The on-board operations scheduling service maintains a command schedule consisting of telecommand packets together with their scheduling attributes. The scheduling attribute of a telecommand indicate the following:

- The sub-schedule with which the telecommand is associated. A sub-schedule is a grouping mechanism for telecommands that enables them to be controlled together to others in the same group.
- The absolute on-board time at which the telecommand packet is released to its destination application process.

Absolute times shall be expressed in OBT reference for the on-board operations scheduling service.

Telecommand release status

The on-board operations scheduling service shall maintain appropriate information to determine whether a telecommand should be released or not at its due time.

The release status of a telecommand shall be affected by the user requests to enable or disable the release of all or a subset of the telecommands in the command schedule. The telecommand release status shall be either “disabled” or “enabled”.

The release status of a telecommand shall be “enabled” if the release of telecommands is enabled from the command schedule, from the sub-schedule to which the telecommand belongs or from the destination application process of the telecommand.

The release status shall be “disabled” in all other cases.

Conceptually, this is as if each telecommand has three independent controlling attributes (at schedule level, at sub-schedule level and at destination application process level) whose values determine the release status of the telecommand in accordance with Table 7.11 1.

Schedule	Sub-schedule	Destination application process	Release status
D(isabled)	E(nabled)	E	D
D	D	E	D
D	E	D	D
D	D	D	D
E	E	E	E
E	D	E	D
E	E	D	D
E	D	D	D

Table 7.11-A : Decision table for the release status of a telecommand

The scheduling activity

The processing of a telecommand packet whose release time is due shall always be performed (even if the command schedule is disabled).

The corresponding service activity shall be:

a. The telecommand shall not be released if the telecommand release status is “disabled”

b. Otherwise, the telecommand shall be released. Where applicable, its execution result and the time of its scheduling event shall be determined.

The MTL can schedule commands in two different modes:

- Absolute Time Scheduling: commands are scheduled according to the on-board absolute time (OBT).
- Orbit Position Scheduling: commands are scheduled according to the orbit position by applying a time offset to the OBT. This time offset corresponds to the difference between the planned ascending crossing time and the one measured on-board.

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(11,1)	Enable release of telecommands	DMS
TC(11,2)	Disable release of telecommands	DMS
TC(11,3)	Reset command schedule	DMS
TC(11,4)	Insert telecommands in command schedule	DMS
TC(11,5)	Delete telecommands	DMS
TC(11,6)	Delete telecommands over time period	DMS
TM(11,10)	Detailed Schedule report	DMS
TC(11,11)	Detailled reporting of the Command Schedule over time period	DMS
TM(11,13)	Summary Schedule report	DMS
TC(11,14)	Summary reporting of command schedule over time period	DMS
TC(11,129)	Set Scheduling mode	DMS

Table 7.11-B : Service 11 sub-services

Note

Since the total maximum size of a TC is limited to 1024 bytes, the TC length (including header and PEC) for commands to be inserted into the MTL can not be larger than 1024 – packet header – 2ndary header – Sub Schedule ID - N – TT - PEC = 1024 - 6 – 4 - 1 -1 - 6 - 2 = 1004 bytes. This results in a maximum packet length for time tagged commands of 1004 - 6 - 1 = 997 bytes.

7.12 SERVICE 12 : ON-BOARD MONITORING

Objective

The on-board monitoring service provides the capability to monitor on-board parameters with respect to checks defined by the ground system and reports any check transitions to the service user. Optionally, an event report may be generated as the result of a given monitoring violation. To achieve this, the service maintains a monitoring list and checks parameter samples according to the information contained therein.

Description

General

A monitoring list shall be maintained which contains the parameter monitoring information, drives the parameter monitoring activity and the generation of check transition reports.

The ground segment can modify or report the contents of the monitoring list using service requests to:

- reset the monitoring list;
- add parameters to, or delete parameters from, the monitoring list;
- enable or disable the monitoring of parameters in the monitoring list;
- report the monitoring information for all parameters in the monitoring list;
- report the set of parameters which are currently out-of-limits..

The ground system can also modify attributes of the on-board monitoring service which determine:

- Whether the monitoring of parameters is enabled or disabled at service level.
- The maximum reporting delay for the check transition report. A check transition report should be issued with no greater delay than this after a new check transition has occurred.

The value of this parameter has an impact both on the average number of check transitions reported in a given check transition report and on the resolution with which the times of reported check transitions are known on the ground (if the reported check transitions are not individually time-stamped).

The monitoring list

The on-board monitoring service shall maintain static monitoring information for each parameter to be monitored, which is provided by the ground system by means of service requests. The parameter monitoring information shall specify:

- the identification of the on-board parameter to be monitored;
- whether the monitoring of the parameter is enabled or disabled;
- the associated validity parameter (if any); this is a Boolean parameter whose value determines whether the parameter is monitored;
- the monitoring interval for the parameter.

The parameter monitoring information shall also include a set of check definitions. A check definition shall provide the information to check a sample of the parameter against either one pair of limits or one expected value. More than one check definition can be associated with a given parameter.

A check definition shall indicate:

- The nature of the check to be performed. This can be a limit-check or an expected-value-check.
  - For a limit-check, a low-limit value and a high-limit value shall be specified.
  - For an expected-value-check, an expected value shall be specified.
- A "number of repetitions (#REP)" which indicates the number of successive samples of the parameter that fail (or succeed) the check before establishing a new checking status for the parameter.
- The identifier (EID) of an event report that shall be generated if the corresponding check fails. Severity of the event report is determined by the EID according to §7.5.

Checking activity and check state

The on-board monitoring service shall maintain a check state corresponding to each check definition for each parameter to be monitored.

The check state shall include information about the previous and current checking status of the parameter for the given check definition and the time at which the transition to that checking status occurred. This information shall be downlinked when the ground system requests a report of the parameters which are currently out-of-limit.

A check definition shall be "enabled" and used for checking a parameter when all the following conditions are set:

- the monitoring of parameters is enabled at service-level,
- the monitoring of the parameter is enabled,
- the parameter is valid (check validity parameter value = "TRUE")

Otherwise the check definition shall be "disabled" and shall not be used for checking the parameter

Whenever a sample of the parameter is available for checking, the service shall perform the following checking activity independently for each parameter check definition (and update its check state accordingly):

- If the check definition is "disabled" then the new checking status shall immediately become either "Unchecked" or "Invalid" depending on whether the checking of the parameter is disabled or the parameter. By default, the initial current checking status of a parameter with respect to the check definition shall be "Unchecked" when the parameter is added to the monitoring list or when a new check definition for the parameter is added at a later time.
- If the check definition is "enabled" then the parameter sample is a valid sample for checking. It shall be checked against the limit pair (or expected value) if sufficient consecutive valid samples have been accumulated. For a limit-check or expected-value-check, if the last #REP successive valid samples of the parameter (including the current one) have consistently failed (or consistently passed) the check, then the parameter shall be assigned a new checking status. The new checking status shall be equal to the result of the check of the current sample, i.e. either "Below low limit", "Above high limit", "Within limits", "Unexpected value" or "Expected value". When the previously determined checking status of a parameter with respect to a limit-check was "Within limits", and when successive samples are alternately "Below low limit" and "Above high limit", these earn the parameter a new checking status. However, when the last known checking status was "Above high limit", then sequence of consecutive "Below low limit" samples shall be available before a new checking status is assigned.
- Having elaborated a new checking status for the parameter, a comparison between the previous and new checking statuses shall be performed. If they differ, then a check transition shall be recorded (conceptually this is recorded in a transition reporting list).
- When a check transition is detected, the transition time shall be recorded in the corresponding check state. If a check transition occurs for which the check definition identifies an associated event report shall be generated, containing the specified report identifier (EID) with auxiliary parameters that are the ones recorded in the transition report. The check transitions concerned are those where the checking status changes to "Below low limit", "Above high limit", "Unexpected value", (depending on the parameter and check type) where it was previously something different.
- On ASTROSAT-250, transitions to IN-LIMIT or EXPECTED-VALUE are recorded as well in the transition table.

The current checking status and associated transition times can be reported to the ground system on request.

The transition reporting list

During the course of the monitoring activity, an ordered list of checking status transitions shall be established. Within this list, there can be more than one checking status transition for a given parameter (e.g. transitions relating to different check definitions; transitions corresponding to different samples of the parameter).

Each checking status transition in the list shall be characterized by:

- the parameter for which the checking status transition was detected;
- the value of the parameter at the time the checking status transition was detected;
- the type of the transition, defined by the previous and the new checking statuses;
- the value of the limit or expected value which was crossed or violated.

The transition reporting list shall be downlinked via a check transition report no later than the maximum reporting delay after the time of the first transition in the list. The list shall be cleared after downlink

Auxiliary information

It is assumed that the on-board monitoring service has access to other information used for the detection of errors in the processing of service requests. This includes the following:

- The maximum number of entries of the monitoring list.
- The list of parameters which can be accessed, and can thus be monitored, by the application process.
- The type(s) of check for each parameter which can be monitored (limit-check or expected-value-check).
- The list of Boolean on-board parameters which can be accessed by the application process and can be used as validity parameter.

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(12,1)	Enable monitoring of parameter	All PIDs
TC(12,2)	Disable monitoring of parameter	All PIDs
TC(12,3)	Change Maximum reporting delay	All PIDs
TC(12,4)	Clear the monitoring list	All PIDs
TC(12,5)	Add parameter to monitoring list	All PIDs
TC(12,6)	Delete parameters from monitoring list	All PIDs
TC(12,8)	Report current monitoring list	All PIDs
TM(12,9)	Current monitoring list Report	All PIDs
TC(12,10)	Report current parameter out-of-limit list	All PIDs
TM(12,11)	Current Parameters Out-of-limit list Report	All PIDs
TM(12,12)	Check transition Report	All PIDs

Table 7.12-A : Service 12 sub-services

7.13 SERVICE 13 : LARGE DATA TRANSFER (NOT USED)

Service not included in current AS-250 baseline.

7.14 SERVICE 14 : PACKET FORWARDING CONTROL

Objective

The packet forwarding control service provides the capability to control the forwarding to the ground of telemetry source packets issued by on-board services.

Description

On ASTROSAT-250 this service is provided by a centralised application process which is responsible for routing packets on the downlink (PID = 10, CSW DMS application).

The packet forwarding control service maintains the knowledge of which packets can be transmitted to the ground system per application process.

For a given application process, the forwarding of packets can be "enabled" and "disabled" at the level of:

- a type of packet;
- a subtype of packet;
- a housekeeping packet definition, a diagnostic packet definition or an event report definition.

The forwarding of packets with a given type and subtype shall be "enabled" if and only if the packet type and the packet subtype are both enabled (i.e. if the type is in the set of enabled types and the subtype is in the set of enabled subtypes for that type).

In addition, the forwarding of housekeeping (or diagnostic or event report) packets shall be "enabled" if and only if the packet type, the packet subtype and the housekeeping packet definition (or the diagnostic packet definition or the event report definition) are all enabled.

Conceptually, this is as if each such packet definition has three independent controlling attributes (at type level, at subtype level and at packet structure identification level) whose values determine the forwarding status of the packets in accordance with Table 7.14 1.

Type	Sub-Type	Identification (SID/EID)	Forwarding status
D(isabled)	E(nabled)	E	D
D	D	E	D
D	E	D	D
D	D	D	D
E	E	E	E
E	D	E	D
E	E	D	D
E	D	D	D

Table 7.14-A : Decision table for the forwarding status of a packet

The packet forwarding control service shall maintain the forwarding status for all types and subtypes of packet (per application process) and shall report the list of enabled packets to the ground system on request.

Similarly, the packet forwarding control service shall maintain the forwarding status for all housekeeping packet definitions, diagnostic packet definitions and event report definitions (per application process) and shall report the list of enabled packets to the ground system on request.

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(14,1)	Enable Forwarding of telemetry Source Packets	DMS
TC(14,2)	Disable Forwarding of telemetry Source Packets	DMS
TC(14,3)	Report Telemetry Source Packet Forwarding Status	DMS
TM(14,4)	Telemetry Source Packet Packets Forwarding Status Report	DMS
TC(14,5)	Enable Forwarding of Housekeeping Packets	DMS
TC(14,6)	Disable Forwarding of Housekeeping Packets	DMS
TC(14,7)	Report Enabled Housekeeping Packets	DMS
TM (14,8)	Enabled Housekeeping Packets Report	DMS
TC(14,9)	Enable Forwarding of Diagnostic Packets	DMS
TC(14,10)	Disable Forwarding of Diagnostic Packets	DMS
TC(14,11)	Report Enabled Diagnostic Packets	DMS
TM(14,12)	Enabled Diagnostic Packets Report	DMS
TC(14,13)	Enable Forwarding of Event Report Packets	DMS
TC(14,14)	Disable Forwarding of Event Report Packets	DMS
TC(14,15)	Report Event Report Packets forwarding status	DMS
TM(14,16)	Event Report Packets Forwarding status Report	DMS

Table 7.14-B : Service 14 sub-services

7.15 SERVICE 15 : ON-BOARD STORAGE AND RETRIEVAL

Objective

The ASTROSAT-250 on-board storage and retrieval service (OBSR) is the central service to selectively store the service reports which are generated by all on-board applications in order to give the ground system the capability to request the downlink of the stored data.

On ASTROSAT-250, the packet selection sub-services which select which telemetry source packets are sent for storage and the storage maintenance and downlink sub-services, are combined in the DMS Application of the OBC application SW.

For ASTROSAT-250, the storage device is the so-called "HK mass memory" and provides a storage capacity of 4Gbit on each OBC. The storage of all Packet Stores (PS) is performed in both Mass Memories (OBC A & B) in parallel. Both Mass Memories are completely managed in mirror : storage, deletion and formatting. This is to ease the recovery of deferred TM after a PM reconfiguration.

The on-board storage and retrieval service consists of three parts:

- packet selection sub-services for selection and transfer of telemetry source packets for storage in different packet stores;
- down-link sub-services for playback of telemetry source packets from packet stores.
- storage area maintenance sub-services.

A packet can be stored the OBC and downlinked from the mass memory independently of its sending in the real-time telemetry.

Description

For missions aimed by ASTROSAT-250, with intermittent coverage, packets of high operational significance, instrument and payload HK data which are generated during a period of non-coverage, are stored in a dedicated packet store, such as that they can be selectively dumped during the next period of coverage. One or more packet types and subtypes generated by one or more application processes can be selected for storage in a given packet store.

A packet store is uniquely identified by a "Store ID". The Store ID value 0 stands for 'All packet stores' and can not be used to identify a specific packet strore. The definition of the storage selection used by a given packet selection sub-service can either be predefined or changeable by the ground system. Packets are stored according to their sequence of arrival at the storage and retrieval sub-service. By design each TM source packet is time stamped by the generating application.

The ASTROSAT-250 OBSR service allows to define a maximum of 20 packet stores. The allocation and the usage of the various Packet Stores will be defined by each ASTROSAT-250 mission in the space to ground ICD.

Telemetry source packets stored in a packet store shall be dumped on request. The downlink sub-services allow specification of the starting point, of the amount of data to be down-linked in order to optimally support the data transfer

Each packet store provides the following capabilities:

- one write pointer to indicate the position resp. filling of the packet store
- parallel writing of data to and down-linking of data from the packet store
- an attribute, which defines whether the storage strategy is circular or bounded
- definition of the storage size and start position for data storage

The attributes of each store are accessed by the storage and retrieval sub-service which shall indicate:

- whether the storage strategy is circular or bounded
- the filling of the packet store by means of write and playback pointers
- the maximum size of the packet store;

When a circular packet store is full, any subsequently received packet shall overwrite partly or fully the oldest packet(s) in the list.

When a bounded packet store is full, any subsequently received packet shall be ignored.

It is possible to delete the content of a Packet Store or a group of Packet Stores, up to a specified storage time.

In addition to housekeeping data that can be reported on a regular basis, catalogue information are maintained by the storage and retrieval service for each packet store, which shall be reported to ground on request. This catalogue information shall include:

- identification of storage configuration,
- the percentage of filling of the packet store
- The dates of the first and the last telemetry packet of the packet store.

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(15,1)	Enable storage in specified packet stores	DMS
TC(15,2)	Disable storage in specified packet stores	DMS
TC(15,3)	Add packets to storage selection definition	DMS
TC(15,4)	Remove packets from storage selection definition	DMS
TC(15,5)	Report storage selection definition	DMS
TM(15,6)	Storage selection definition report	DMS
TC(15,9)	Downlink packet store content for time period	DMS
TC(15,10)	Delete Packet Stores content	DMS
TC(15,11)	Delete packet stores content up to specified Storage Time	DMS
TC(15,128)	Format HK mass memory	DMS
TC(15,129)	Report HK mass memory format	DMS
TM(15,130)	HK mass memory format report	DMS
TC(15,131)	Stop downlink of Packet Stores content	DMS
TC(15,132)	Start downlink of Packet Stores content	DMS
TC(15,133)	Add SID's to storage selection definition	DMS
TC(15,134)	Remove SID's from storage selection definition	DMS
TC(15,135)	Report SID storage selection definition	DMS
TM(15,136)	SID's storage selection definition Report	DMS
TC(15,137)	Change packet store attributes	DMS

Table 7.15-A : Service 15 sub-services

7.16 SERVICE 16 : NOT DEFINED

7.17 SERVICE 17 : TEST

For ASTROSAT-250, the spacecraft bus architecture makes use of MIL-STD-1553. This type of bus allows the health status monitoring of connected remote terminals as cyclic acquisition are made on them (as well as it provides an 'are you alive' capability). To that respect, the functionalities of service 17 for ASTROSAT-250 equipments are already covered without the need of a dedicated service and telecommands described hereafter only apply to the central software.

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(17,1)	Perform Connection Test	All PIDs
TM(17,2)	Link Connection Report	All PIDs

Table 7.17-A : Service 17 sub-services

7.18 SERVICE 18 : ON BOARD OPERATIONS PROCEDURE

Objective

The ground system can define a set of operation procedures (OBCP) that can be loaded to an application process, which then manages the on-board storage of these procedures and their subsequent execution under ground system control. In principle, such an operations procedure can also be controlled (e.g. started) autonomously on-board, e.g. as the result of detection of a specific on-board event.

The on-board operation procedure service provides standard service requests and reports for controlling the execution of these procedures and monitoring their status.

Description

An on-board operation procedure managed by a given application process is uniquely identified by a "Procedure ID".

An on-board operations procedure shall be composed of a number of steps. Each step is identified by a unique "Step ID" and is made of 2 parts : the TC to be executed and the delay to wait before starting the execution of the TC. This delay is relative to the start of execution of the previous TC, or the start of the procedure if it is the first step.

A set of control commands is defined to enable control over the procedure flow and is provided as sub-service of the OBCP service by dedicated private service sub-types. The following OBCP directives ensure provision of the requested essential OBCP capability set:

- OBCP LOGICAL DECISION DIRECTIVE: branches the execution of a running OBCP to a step, based on a test of an on-board TM parameter from the on-board data pool.
- OBCP JUMP DIRECTIVE: branches the execution of a running OBCP to another step.
- OBCP SEND EVENT DIRECTIVE: generates an event report reporting the current OBCP and step identifiers, allowing establishing a full trace of the OBCP.An on-board operations procedure shall have an execution status which indicates whether it is currently "inactive" or "active".

The service shall maintain a list of the currently loaded Procedures and a list of the currently active Procedures and these lists shall be reported to the ground on request.

Reporting on the progress of execution of a Procedure can be achieved either by the use of telecommand verification reports or event reports.

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(18,2)	Delete Procedure	DMS
TC(18,3)	Start Procedure	DMS
TC(18,4)	Stop Procedure	DMS
TC(18,8)	Report List of Onboard Operation Procedures	DMS
TM(18,9)	On-board Operation Procedures List Report	DMS
TC(18,128)	Add/Replace TC to OBCP	DMS
TC(18,129)	Delete TC from OBCP	DMS
TC(18,130)	Dump Onboard Procedure	DMS
TM(18,131)	Onboard Procedure Dump	DMS
TC(18,140)	OBCP LOGICAL DECISION DIRECTIVE	DMS
TC(18,141)	OBCP JUMP DIRECTIVE	DMS
TC(18,142)	OBCP SEND EVENT DIRECTIVE	DMS

Table 7.18-A : Service 18 sub-services

7.19 SERVICE 19 : EVENT / ACTION SERVICE

Objective

As an extension to the on-board capability for detecting events and reporting them asynchronously to the ground system, this service provides the capability to define an action that is executed autonomously on-board when a given event is detected. The class of events that can give rise to an action are those that also give rise to an event report and the associated action can be a telecommand of any standard type (i.e. as defined in this Standard) or any mission-specific telecommand.

Description

The service shall maintain a list of events to be detected that contains the following information:

- Process ID generating the event report;
- Event report ID;
- Associated action (telecommand packet);
- Status of the action – enabled or disabled;

The list shall be updated in accordance with requests from ground and the list information shall be reported to ground on request. The service can detect event reports (TM(5,[1-4]) generated by all application process. On reception of an event report, the service shall scan the detection list and if a matching event report is detected and the associated action is enabled, the corresponding telecommand packet shall be sent to the destination application process.

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(19,1)	Add an Event to the Detection List	DMS
TC(19,2)	Delete an Event from the Detection List	DMS
TC(19,3)	Clear the Events Detection List	DMS
TC(19,4)	Enable Actions	DMS
TC(19,5)	Disable Actions	DMS
TC(19,6)	Report the Event Detection List	DMS
TM(19,7)	Event detection List Report	DMS

Table 7.19-A : Service 19 sub-services

7.20 SERVICE 140 : PARAMETER MANAGEMENT

Objective

This service allows the ground to manage on board parameters by changing or reading onboard parameters values.

Description

An onboard parameter allow the ground to access to CSW data that can be either software variable or equipment acquisition data.

The onboard parameter function manages an onboard parameters list per application process ID. Onboard parameters list definition is extracted from the SDB.

Each onboard parameter contains the following information:

- the onboard parameter identifier
- the onboard parameter address
- the onboard parameter length
- the onboard parameter type

Ground commanding

The onboard parameters definition is defined in the SDB and is supposed to be frozen for a flight software release. Nevertheless, a subset of parameters are tagged with a flag 'spare' in order to provide to ground the facility to modify these parameters definition. This can be useful for ground investigations in order to be able to easily handle through diagnostic TM, SW variables that were not initially defined in the SDB because no necessary for nominal ground operations. This service is supported by TC(140,4).

In order not to resort to load and dump services which are heavy and risky to use, the service 140 provides access to onboard parameters through their parameter id rather than their address. This may very useful to upload new value in a parameter when a private TC doesn't exist to do that..

Sub-services

Subtype	Sub-service Name	CSW applicability
TC(140,1)	Set N parameters	All PIDs
TC(140,2)	Get N parameters	All PIDs
TM(140,3)	Parameter Report	All PIDs
TC(140,4)	Define New Onboard parameter	All PIDs

Table 7.20-A : Service 140 sub-services

8 APPENDIXES

8.1 APPENDIX 1 - PRIVATE SERVICE TYPES ASSIGNMENT

The table below allocates a range to each application for its private services.

Private Service Types	Allocated Application Process
0-127	Reserved for standard services
128-139	Reserved (not used)
140	Parameter management
141	Functional monitoring
142	Dwell
143 to 149	Reserved (not used)
150	OBC management
151 to 153	Reserved (not used)
154 to 159	System / DMS
160 to 169	AOCS
170 to 179	Payload control
180 to 189	Platform/Bus/Thermal control
190 to 256	Reserved (not used)

Table 8.1-A : Private service type ranges

8.2 APPENDIX 2 - APPLICATION PROCESS ID ASSIGNMENT

The following table provides the global allocation of APIDs to the various users within ASTROSAT-250.

The Application Process ID (APID) is structured into two fields:

The least significant 4-bits within the APID form a field called «Packet Category». This field identifies different categories of TM packets, which the ground typically processes in different ways and for which a separate accounting is required to be kept. For telecommands this field should be set to decimal '12' (PRIVATE).

The most significant 7 bits of the APID form a field called «Process ID». This field identifies in general terms the process generating the TM packet (or which the TC packet is addressed to). For example a unit on-board can be assigned many Process IDs and use them to define different packets of the same category from different processes running in the same unit.

The tables below show the applicable Process IDs and Packet Categories assignment. The TM/TC Packets Table in Appendix 1 maps the Packet Categories to Service Types (for TM packets). The number in the table below are in decimal value.

Process ID (PID) Allocation	
0	TIME packet
1 to 8	(spare)
9	OBC hardware
10	CSW DMS application
11	CSW AOCS application
12	CSW Payload application
13	CSW Platform application
14	CSW System application
15 to 100	(spare)
101 to 111	Reserved for ESOC
112 to 126	SCOE (AIT)
127	IDLE packet

Table 8.2-A : Process Identification (PID) Allocation

The Packet Category is assigned as follows:

Packet Category (PCAT)	Meaning
0	TIME
1	ACKNOWLEDGE
2	HK_RT[1]
3	TABLE
4	HK_PB
5 (not used)	Spare
6	Spare
7	EVENT
8	DIAGNOSTIC
9	DUMP
10 (not used)	Spare
12	TELECOMMAND
13 (not used)	Spare
14	Reserved for OCC/EGSE
15	IDLE

Table 8.2-B : Category (PCAT) Allocation

[1] 2 categories are used in service 3, allowing to distinguish packets sent to the Real Time VC (HK_RT) from the ones stored on-board and downlinked in the Playback VC (HK_PB), and therefore avoiding sequence counters gaps in the real-time telemetry flow when a packet is stored on-board.

8.3 APPENDIX 3 - CYCLIC REDUNDANCY CODE (CRC)

8.3.1 Specification

The Packet Error Control Field is a 16-bit field, which occupies the two trailing bytes of a TC Packet.

The purpose of this field is to provide a capability for detecting errors which may have been introduced into the frame by the lower protocol layers during the transmission process and may have remained undetected.

The standard error detection encoding/decoding procedure, which is described in detail in the following paragraphs, produces a 16 bit Packet Check Sequence (PCS) which is placed in the Packet Error Control Field.

This code is intended only for error detection purpose and shall not be used for error correction.

The characteristics of the PCS are those of a cyclic redundancy code (CRC) and are generally expressed as follows:

a) The generator polynomial is G(x) = X16 + X12 + X5 + 1

b) Both encoder and decoder are initialised to the "all-ones" state for each Packet.

c) PCS generation is performed over the entire Packet including the Packet Header less the final 16-bit PCS.

d) The code has the following capabilities when applied to an encoded block of less than 32768 bits (215 bits) :

- All error sequences composed of an odd number of bit errors will be detected
- All error sequences containing two bit errors anywhere in the coded block will be detected
- If a random error sequence containing an even number of bit errors (greater than or equal to four) occurs within the block, the probability that the error will be undetected in approximately 2-15 (or 3 x 10-5).
- All single error bursts spanning 16 bits or less will be detected provided no other errors occur within the block.

For blocks longer than 4096 bits, the specified performance cannot be guaranteed.

8.3.2 Encoding Procedure

The encoding procedure accepts an (n-16)-bit message and generates a systematic binary (n, n-16) block code by appending a 16-bit Packet Check Sequence (PCS) as the final 16 bits of the block. This PCS is inserted into the Packet Error Control Field. The equation for PCS is:

PCS = [X16. M (X) + X (n-16). L (X)] MODULO G (X)

Where

M (X) is the (n-16)-bit message to be encoded expressed as a polynomial with binary coefficients, n being the number of bits in the encoded message (i.e. the number of bits in the complete Packet).

L (X) is the pre-setting polynomial given by:

L (X) = S 15i=0 Xi (all "1" polynomial of order 15)

G (X) is the CCITT Recommendation V.41 generating polynomial given by:

G (X) = X16 + X12 + X5 + 1

Where + is the modulo 2 addition operator (exclusive OR)

Note that the encoding procedure differs from that of a conventional cyclic block encoding operation in that the X (n-16). L (X) term has the effect of presenting the shift register to an all ones state (rather than a conventional all zeros state) prior to encoding.

8.3.3 Decoding Procedure

The error detection syndrome, S (X) is given by

S (X) = (X16 . C*(X) + Xn . L (X) MODULO G (X)

Where C*(X) is the received block in polynomial form.

S (X) is the syndrome polynomial which will be zero if no error has been detected.

8.3.4 Verification of Compliance

The binary sequences defined in this section are provided to the designers of packet systems as samples for testing and verification of a specific CRC error detection implementation.

All data are given in hexadecimal notation. For a given field (data or CRC), the left most hexadecimal character contains the most significant bit (i.e. bit 0 of the CCSDS convention).

DATA	Packet Check Sequence (CRC)
00 00	1D 0F
00 00 00	CC 9C
AB CD EF 01	04 A2
14 56 F8 9A 00 01	7F D5

8.4 APPENDIX 4 - STANDARD SPACECRAFT TIME SOURCE PACKET

The Standard Spacecraft Time Source Packet shall be used to transport both the regular Spacecraft Elapsed Time samples and On Board Time samples to ground for time correlation with UTC by the ground segment during periods of ground contact. Its structure is defined in [AD-03] and it is shown in Figure 7.21 1 below.

SOURCE PACKET HEADER (48 bits)							PACKET DATA FIELD (136 bits)		
PACKET ID				PACKET SEQUENCE CONTROL		PACKET LENGTH	S-FIELD	P-FIELD	T-FIELD
Version Number	Type	Data Field Header Flag	Application Process ID	Segment-ation Flags	Source Sequence Count				
3	1	1	11	2	14				
16				16		16	8	8	56

Figure 7.21-C : Standard Spacecraft Time Source Packet Fields

The time carried by the T-field of the packet shall relate to the instant of occurrence of the leading edge of the first bit of the attached synchronisation marker of the telemetry transfer frame of virtual channel "0" with a virtual channel frame count of "0" as recorded in OBC Spacecraft Elapsed Time (SCET).

The field contents of the Standard Spacecraft Time Source Packet header and data field are specified below:

Packet ID

Version Number :

The version number must be set 000BIN.

Type :

The type must be set to zero indicating telemetry source packet.

Data Field Header Flag :

The data field header flag must be set to zero. No data field header.

Application Process ID :

The Application process ID shall be set to all zeros.

Sequence Control

Segmentation [Grouping] Flags

The segmentation flags must be set to "11" for "stand alone" packet.

Source Sequence Count

The source sequence count of the time packet must be incremented by 1 whenever the source releases a packet. Ideally, this counter should never re-initialise, however, under no circumstances shall it "short-cycle" (i.e. have a discontinuity other than to a value zero).

The counter wraps around from 214 -1 to zero.

Packet Length

The packet length field specifies the number of bytes contained within the Packet Data Field. The number is an unsigned integer "C" where:

C = (Number of bytes in Packet Data Field) - 1

In this case, the number of bytes is sixteen (i.e. C=15).

It should be noted that the actual length of the entire Standard Spacecraft Time Source Packet, including the Packet Header, is 6 bytes longer.

S-Field

Bits 0 through 3 are not used and must be set to zeros.

Bits 4 through 7 define the time packet generation rate (once every 2expRate telemetry transfer frames on VC0).

Spacecraft Time Source Packet.

P-Field

Must be set to "00101111 BIN " to indicate that the following time format consists of 4 coarse time bytes and 3 fine time bytes. This allows a resolution of approx. 60 ns.

T-Field

This field will contain the Spacecraft Elapsed Time, consistent with the CCSDS Unsegmented Time Code (CUC) format.

Bits 0 through 31 must contain the coarse Spacecraft Elapsed Time as an unsegmented binary count of seconds.

Bits 32 through 55 must contain the fine Spacecraft Elapsed Time as an unsegmented binary power of subseconds.

8.5 APPENDIX 5 - IDLE PACKET STRUCTURE

The idle packet will be used to fill the telemetry transfer frame when a frame has to be transmitted and an insufficient number of source packets are available to complete the transfer frame. This may be the case when the source data rate is low compared to the frame period. Its structure is as shown in Figure 7.21 2 below.

SOURCE PACKET HEADER (48 bits)							**PACKET DATA FIELD (VARIABLE)**
PACKET ID				PACKET SEQUENCE CONTROL		PACKET LENGTH	FILLER PATTERN
Version Number	Type	Data Field Header Flag	Application Process ID	Segment-ation Flags	Source Sequence Count		
3	1	1	11	2	14		
16				16		16	Variable

Figure 8.5-A : Idle Packet Fields

The field contents of the Idle Packet header and data field are specified below:

Packet ID

Version Number

The version number must be set to 000BIN.

Type

The type must be set to zero.

Data Field Header Flag

The data field header flag must be set to zero.

Application Process ID

The Application process ID must be set to all ones.

Sequence Control

Segmentation [Grouping] Flags

The segmentation flags must be set to "11".

Source Sequence Count

The source sequence count of the idle packet must be set to zero (0).

Packet Length

The packet length field specifies the number of bytes contained within the Packet Data Field. The number is an unsigned integer "C" where:

C = (Number of bytes in Packet Data Field) - 1

The length of the packet may be freely chosen by the user. It should be noted that the actual length of the entire Idle Packet, including the Packet Header, is 6 bytes longer.

Filler Pattern

The content of the Idle Packet data field shall be random data.

DISTRIBUTION LIST

	Corporate use	Project Limited
Internal Classification		X

Controlled	Configured
	X

AS250 Avionics Team	Distribution
BARADAT Pierre	X
BISCARROS Didier	X
CARLIER Corinne	X
CAZALS Serge	X
COSCULLUELA Valerie	
DEFENDINI Ange	X
FOURNE Brigitte	
GILLOT Bernard	X
LEBLOND Philippe	X
LECUREUX Yves	
AS250 SW team	
ALBERT Pierre	X
DIRIS Philippe	X
LEGER Béatrice	X
MECHIN Xavier	X
Astrium AstroTerra Project	
BELLEAU Patrick	X
BOUSQUET Christophe	X
HAMEURY Olivier	X
ECE Seosat Project	
DE MIGUEL Elena Diana	X
MONTES BAZAN Tamara	X
SANZ CUESTA Juan Andrés	X

AS250 Avionics Team	Distribution
LEBREDONCHEL Jérôme	X
MARCILLE Herve	X
MARTINEZ Alain	X
PENALVA Sonia	
POQUET GOURDON Emmanuelle	
SHAABAN Sami	X
THIEUW Alain	
AS250 Documentation manager	X
AS250 Configuration manager	X
Astrium MPC	
BEUGNET Laurent	X
CORREGE Alain	X
DEBUS Volker	X
Astrium KRS Project	
PAGES Christophe	X
GELIS Daniel	X
HERRGOTT Gilles	X