

CORDET FRAMEWORK - PUS EXTENSION

TM/TC Interface Control Document

P&P Software GmbH
High Tech Center 1
8274 Tägerwilen (CH)

Web site: www.pnp-software.com
E-mail: pn-p-software@pn-p-software.com

Written By:	Alessandro Pasetti
Checked By:	n.a.
Document Ref.:	PP-IC-PUX-0001
Issue:	0.2
Created On:	20/05/2019, at: 22:40

CORDET FRAMEWORK - PUS EXTENSION

TM/TC Interface Control Document

P&P Software GmbH
High Tech Center 1
8274 Tägerwilen (CH)

Web site: www.pnp-software.com
E-mail: pn-p-software@pn-p-software.com

Written By:	Alessandro Pasetti
Checked By:	n.a.
Document Ref.:	PP-IC-PUX-0001
Issue:	0.2
Created On:	20/05/2019, at: 22:40

Contents

1	References	8
2	Introduction	9
3	Command and Report Packet Structure	11
4	Service Overview	13
5	Detailed Definition of PUS Services	17
5.1	TM(1,1) SuccAccRep	18
5.2	TM(1,2) FailedAccRep	19
5.3	TM(1,3) SuccStartRep	20
5.4	TM(1,4) FailedStartRep	21
5.5	TM(1,5) SuccPrgrRep	22
5.6	TM(1,6) FailedPrgrRep	23
5.7	TM(1,7) SuccTermRep	24
5.8	TM(1,8) FailedTermRep	25
5.9	TM(1,10) FailedRoutingRep	26
5.10	TC(3,1) CreHkCmd	27
5.11	TC(3,2) CreDiagCmd	28
5.12	TC(3,3) DelHkCmd	29
5.13	TC(3,4) DelDiagCmd	30
5.14	TC(3,5) EnbHkCmd	31
5.15	TC(3,6) DisHkCmd	32
5.16	TC(3,7) EnbDiagCmd	33
5.17	TC(3,8) DisDiagCmd	34
5.18	TC(3,9) RepStructHkCmd	35
5.19	TM(3,10) RepStructHkRep	36
5.20	TC(3,11) RepStructDiagCmd	37
5.21	TM(3,12) RepStructDiagRep	38
5.22	TM(3,25) Rep (SID_HK_CNT)	39
5.23	TM(3,25) Rep (SID_N_OF_EVT)	40
5.24	TM(3,26) DiagRep	41
5.25	TC(3,27) OneShotHkCmd	42
5.26	TC(3,28) OneShotDiagCmd	43
5.27	TC(3,31) ModPerHkCmd	44
5.28	TC(3,32) ModPerDiagCmd	45
5.29	TM(5,1) Rep1 (EVT_DOWN_ABORT)	46
5.30	TM(5,1) Rep1 (EVT_DUMMY_1)	47
5.31	TM(5,1) Rep1 (EVT_UP_ABORT)	48
5.32	TM(5,2) Rep2 (EVT_CLST_FULL)	49
5.33	TM(5,2) Rep2 (EVT_DUMMY_2)	50
5.34	TM(5,3) Rep3 (EVT_DUMMY_3)	51
5.35	TM(5,3) Rep3 (EVT_FMON_FAIL)	52
5.36	TM(5,3) Rep3 (EVT_MON_DEL_I)	53
5.37	TM(5,3) Rep3 (EVT_MON_DEL_R)	54
5.38	TM(5,3) Rep3 (EVT_MON_LIM_I)	55
5.39	TM(5,3) Rep3 (EVT_MON_LIM_R)	56
5.40	TM(5,4) Rep4 (EVT_DUMMY_4)	57
5.41	TC(5,5) EnbCmd	58

5.42	TC(5,6)	DisCmd	59
5.43	TC(5,7)	RepDisCmd	60
5.44	TM(5,8)	DisRep	61
5.45	TC(11,1)	EnbTbsCmd	62
5.46	TC(11,2)	DisTbsCmd	63
5.47	TC(11,3)	ResTbsCmd	64
5.48	TC(11,4)	InsTbaCmd	65
5.49	TC(11,5)	DelTbaCmd	66
5.50	TC(11,20)	EnbSubSchedCmd	67
5.51	TC(11,21)	DisSubSchedCmd	68
5.52	TC(11,22)	CreGrpCmd	69
5.53	TC(11,23)	DelGrpCmd	70
5.54	TC(11,24)	EnbGrpCmd	71
5.55	TC(11,25)	DisGrpCmd	72
5.56	TC(11,26)	RepGrpCmd	73
5.57	TM(11,27)	GrpRep	74
5.58	TC(12,1)	EnbParMonDefCmd	75
5.59	TC(12,2)	DisParMonDefCmd	76
5.60	TC(12,3)	ChgTransDelCmd	77
5.61	TC(12,4)	DelAllParMonCmd	78
5.62	TC(12,5)	AddParMonDefCmd	79
5.63	TC(12,6)	DelParMonDefCmd	81
5.64	TC(12,7)	ModParMonDefCmd	82
5.65	TC(12,8)	RepParMonDefCmd	83
5.66	TM(12,9)	RepParMonDefRep	84
5.67	TC(12,10)	RepOutOfLimitsCmd	85
5.68	TM(12,11)	RepOutOfLimitsRep	86
5.69	TM(12,12)	CheckTransRep	87
5.70	TC(12,13)	RepParMonStatCmd	88
5.71	TM(12,14)	RepParMonStatRep	89
5.72	TC(12,15)	EnbParMonFuncCmd	90
5.73	TC(12,16)	DisParMonFuncCmd	91
5.74	TC(12,17)	EnbFuncMonCmd	92
5.75	TC(12,18)	DisFuncMonCmd	93
5.76	TC(12,19)	EnbFuncMonDefCmd	94
5.77	TC(12,20)	DisFuncMonDefCmd	95
5.78	TC(12,21)	ProtFuncMonDefCmd	96
5.79	TC(12,22)	UnprotFuncMonDefCmd	97
5.80	TC(12,23)	AddFuncMonDefCmd	98
5.81	TC(12,24)	DelFuncMonDefCmd	99
5.82	TC(12,25)	RepFuncMonDefCmd	100
5.83	TM(12,26)	RepFuncMonDefRep	101
5.84	TC(12,27)	RepFuncMonStatCmd	103
5.85	TM(12,28)	RepFuncMonStatRep	104
5.86	TM(13,1)	DownFirstRep	105
5.87	TM(13,2)	DownInterRep	106
5.88	TM(13,3)	DownLastRep	107
5.89	TC(13,9)	UpFirstCmd	108
5.90	TC(13,10)	UpInterCmd	109
5.91	TC(13,11)	UpLastCmd	110
5.92	TM(13,16)	UpAbortRep	111

5.93 TC(13,129) StartDownCmd	112
5.94 TC(13,130) AbortDownCmd	113
5.95 TC(17,1) AreYouAliveCmd	114
5.96 TM(17,2) AreYouAliveRep	115
5.97 TC(17,3) ConnectCmd	116
5.98 TM(17,4) ConnectRep	117
5.99 TC(255,1) Sample1	118
5.100TM(1,1) SuccAccRep	119
5.101TM(1,2) FailedAccRep	120
5.102TM(1,3) SuccStartRep	121
5.103TM(1,4) FailedStartRep	122
5.104TM(1,5) SuccPrgrRep	123
5.105TM(1,6) FailedPrgrRep	124
5.106TM(1,7) SuccTermRep	125
5.107TM(1,8) FailedTermRep	126
5.108TM(1,10) FailedRoutingRep	127
5.109TC(3,1) CreHkCmd	128
5.110TC(3,2) CreDiagCmd	129
5.111TC(3,3) DelHkCmd	130
5.112TC(3,4) DelDiagCmd	131
5.113TC(3,5) EnbHkCmd	132
5.114TC(3,6) DisHkCmd	133
5.115TC(3,7) EnbDiagCmd	134
5.116TC(3,8) DisDiagCmd	135
5.117TC(3,9) RepStructHkCmd	136
5.118TM(3,10) RepStructHkRep	137
5.119TC(3,11) RepStructDiagCmd	138
5.120TM(3,12) RepStructDiagRep	139
5.121TM(3,25) Rep (SID_HK_CNT)	140
5.122TM(3,25) Rep (SID_N_OF_EVT)	141
5.123TM(3,26) DiagRep	142
5.124TC(3,27) OneShotHkCmd	143
5.125TC(3,28) OneShotDiagCmd	144
5.126TC(3,31) ModPerHkCmd	145
5.127TC(3,32) ModPerDiagCmd	146
5.128TM(5,1) Rep1 (EVT_DOWN_ABORT)	147
5.129TM(5,1) Rep1 (EVT_DUMMY_1)	148
5.130TM(5,1) Rep1 (EVT_UP_ABORT)	149
5.131TM(5,2) Rep2 (EVT_CLST_FULL)	150
5.132TM(5,2) Rep2 (EVT_DUMMY_2)	151
5.133TM(5,3) Rep3 (EVT_DUMMY_3)	152
5.134TM(5,3) Rep3 (EVT_FMON_FAIL)	153
5.135TM(5,3) Rep3 (EVT_MON_DEL_I)	154
5.136TM(5,3) Rep3 (EVT_MON_DEL_R)	155
5.137TM(5,3) Rep3 (EVT_MON_LIM_I)	156
5.138TM(5,3) Rep3 (EVT_MON_LIM_R)	157
5.139TM(5,4) Rep4 (EVT_DUMMY_4)	158
5.140TC(5,5) EnbCmd	159
5.141TC(5,6) DisCmd	160
5.142TC(5,7) RepDisCmd	161
5.143TM(5,8) DisRep	162

5.144TC(11,1)	EnbTbsCmd	163
5.145TC(11,2)	DisTbsCmd	164
5.146TC(11,3)	ResTbsCmd	165
5.147TC(11,4)	InsTbaCmd	166
5.148TC(11,5)	DelTbaCmd	167
5.149TC(11,20)	EnbSubSchedCmd	168
5.150TC(11,21)	DisSubSchedCmd	169
5.151TC(11,22)	CreGrpCmd	170
5.152TC(11,23)	DelGrpCmd	171
5.153TC(11,24)	EnbGrpCmd	172
5.154TC(11,25)	DisGrpCmd	173
5.155TC(11,26)	RepGrpCmd	174
5.156TM(11,27)	GrpRep	175
5.157TC(12,1)	EnbParMonDefCmd	176
5.158TC(12,2)	DisParMonDefCmd	177
5.159TC(12,3)	ChgTransDelCmd	178
5.160TC(12,4)	DelAllParMonCmd	179
5.161TC(12,5)	AddParMonDefCmd	180
5.162TC(12,6)	DelParMonDefCmd	182
5.163TC(12,7)	ModParMonDefCmd	183
5.164TC(12,8)	RepParMonDefCmd	184
5.165TM(12,9)	RepParMonDefRep	185
5.166TC(12,10)	RepOutOfLimitsCmd	186
5.167TM(12,11)	RepOutOfLimitsRep	187
5.168TM(12,12)	CheckTransRep	188
5.169TC(12,13)	RepParMonStatCmd	189
5.170TM(12,14)	RepParMonStatRep	190
5.171TC(12,15)	EnbParMonFuncCmd	191
5.172TC(12,16)	DisParMonFuncCmd	192
5.173TC(12,17)	EnbFuncMonCmd	193
5.174TC(12,18)	DisFuncMonCmd	194
5.175TC(12,19)	EnbFuncMonDefCmd	195
5.176TC(12,20)	DisFuncMonDefCmd	196
5.177TC(12,21)	ProtFuncMonDefCmd	197
5.178TC(12,22)	UnprotFuncMonDefCmd	198
5.179TC(12,23)	AddFuncMonDefCmd	199
5.180TC(12,24)	DelFuncMonDefCmd	200
5.181TC(12,25)	RepFuncMonDefCmd	201
5.182TM(12,26)	RepFuncMonDefRep	202
5.183TC(12,27)	RepFuncMonStatCmd	204
5.184TM(12,28)	RepFuncMonStatRep	205
5.185TM(13,1)	DownFirstRep	206
5.186TM(13,2)	DownInterRep	207
5.187TM(13,3)	DownLastRep	208
5.188TC(13,9)	UpFirstCmd	209
5.189TC(13,10)	UpInterCmd	210
5.190TC(13,11)	UpLastCmd	211
5.191TM(13,16)	UpAbortRep	212
5.192TC(13,129)	StartDownCmd	213
5.193TC(13,130)	AbortDownCmd	214
5.194TC(17,1)	AreYouAliveCmd	215

5.195TM(17,2) AreYouAliveRep	216
5.196TC(17,3) ConnectCmd	217
5.197TM(17,4) ConnectRep	218
5.198TC(255,1) Sample1	219
6 Data Pool Definition	220
7 Type Definition	229

1 References

The documents referenced in this document are listed in table 1.1.

The following documents are for reference and/or guideline only.

Table 1.1: Referenced documents

Ref	Description	Doc. Number	Iss.
[CR-SP]	The CORDET Framework, www.pnp-software.com/cordetfw	Release	1
[FW-SP]	The Framework Profile, www.pnp-software.com/fwprofile	Release	1.3.1
[PS-SP]	Ground Systems and Operations, Telemetry and Telecommand Packet Utilization Standard	ECSS-E-70-41C	C
[PX-SP]	The PUS Extension of the CORDET Framework – Specification	PP-SP-PUX-001	0.2
[PX-VR]	The PUS Extension of the CORDET Framework – Verification Report	PP-RP-PUX-001	0.2
[PX-UM]	The PUS Extension of the CORDET Framework – User Manual	PP-UM-PUX-001	0.2

2 Introduction

This document is the interface control document for the telemetry reports and telecommands of the PUS Extension of the CORDET Framework. The PUS Extension of the CORDET Framework is specified in reference [CR-SP]. A partial implementation in C is available and its user manual is in reference [CR-UM].

The layout of the commands and reports of the PUS Extension is defined in the CORDET Editor. The CORDET Editor is a proprietary tool of P&P Software GmbH which allows the PUS-compliant commands and reports of a set communicating applications to be specified in terms of:

- The set of supported services
- The set of commands and reports in each service
- The layout of the commands and reports
- The semantics of the commands and reports
- The syntactical type of the parameters in the commands and reports
- The endianness of the representation of the command and report parameters

The CORDET Editor tool includes a suite of generators which can generate the following items:

- Tables listing the services supporting the commands and reports
- Tables listing the commands and reports in each service
- Tables describing the semantics of the commands and reports
- Tables defining the layout of each command or report
- Tables listing the items in the on-board data pool
- Tables describing the service 5 event identifiers
- Tables describing the service 1 command rejection codes
- C-language modules implementing the data pool
- A C-language header file defining the types used for the commands and reports
- C-language modules implementing accessor methods for the parameters of commands and reports

The tables are generated in csv and latex formats. All the tables presented in this document were generated by the CORDET Editor. They are therefore guaranteed to be consistent with the C-language implementation of the PUS Extension.

The command and report interfaces defined in this document are based on the PUS specification of reference [PS-SP]. This specification leaves some choices open concerning, in particular, the syntactical type of the command and report parameters. These choices have been resolved when the commands and reports were defined in the CORDET Editor. These choices are those used for the Test Suite Application of the C-language implementation of the PUS Extension of the CORDET Framework (see section TBD of reference [PX-UM]). This Test Suite Application is one particular instantiation of the PUS Extension of the CORDET Framework. Users who wish to adapt the present document to a different instantiation (e.g. by changing the size of certain command or report parameters) should proceed as follows:

- Modify the definition of the commands and reports within the CORDET Editor
- Run the generator of the CORDET Editor
- Re-compile this document using the newly generated tables for the commands and reports
- Re-compile the framework code using the newly generate C-modules implementing the commands and reports

3 Command and Report Packet Structure

Command and report packets consist of a header, an application part and a packet error control part.

The layouts of the command and report headers (packet headers plus data field header) are defined in tables 3.1 and 3.2. In the CORDET Editor, these layouts can be modified in the "TC Header" and "TM Header" tables.

The Packet Error Control is implemented as a 2-byte CRC. In the CORDET Editor, the size of the CRC is defined in the "extension" menu of the "packet access functions".

A dummy implementation for the CRC is currently provided which sets it to 0xFFFF. The computation of the CRC is one of the adaptation points of the PUS Extension of the CORDET Framework (adaptation point OST-13) and users are expected to modify it to suit their needs.

The maximum length of a command or report packet is given by constant `CR_FW_MAX_PKT_LENGTH` defined in `CrFwUserConstants.h`. Note that, in the Test Suite Application, the maximum packet size has been chosen to be unrealistically small in order to allow convenient verification of situations where a certain piece of information must be spread over multiple packets.

For the APID, the following considerations apply (see also section 3.2 of reference [PX-SP]):

- The APID consists of the concatenation of PCAT and PID
- The PCAT is the same as the CORDET Group attribute
- The PID is the same as the CORDET Application Identifier whose value is defined by constant `CR_FW_HOST_APP_ID` in `CrFwUserConstants.h`

The PCAT is not defined at framework level. Applications define their PCAT implicitly by implenting functions `CrFwPcktSetGroup` and `CrFwPcktGetGroup` in interface `CrFwPckt.h`.

Table 3.1: TC header

Name	Byte	Bit	Size	Value	Description
PcktVersionNmb	0	0	3	0	Packet Version Number
PcktType	0	3	1	0	Packet type flag (1 for command and 0 for report)
SecHeaderFlag	0	4	1	0	Secondary Header Flag
APID	0	5	11	0	Application Process ID (APID) made up of PCAT (7 most significant bits) and PID (4 least significant bits)
SeqFlags	2	0	2	0	Segmentation Flag
SeqCount	2	2	14	0	Source Sequence Counter
PcktDataLen	4	0	16	0	Packet Data Length
PUSVersion	6	0	4	0	PUS Version
AckAccFlag	6	4	1	0	Acknowledge Acceptance Flag
AckStartFlag	6	5	1	0	Acknowledge Start Flag
AckProgFlag	6	6	1	0	Acknowledge Progress Flag

Name	Byte	Bit	Size	Value	Description
AckTermFlag	6	7	1	0	Acknowledge Termination Flag
ServType	7	0	8	0	PUS Service Type
ServSubType	8	0	8	0	PUS Service Sub Type
SrcId	9	0	8	0	Identifier of telecommand source
					Total bits: 80 Total bytes: 10.0 Total words: 5.0

Table 3.2: TM header

Name	Byte	Bit	Size	Value	Description
PcktVersionNmb	0	0	3	0	Packet Version Number
PcktType	0	3	1	0	Packet type flag (1 for command and 0 for report)
SecHeaderFlag	0	4	1	0	Secondary Header Flag
APID	0	5	11	0	Application Process ID (APID) made up of PCAT (7 most significant bits) and PID (4 least significant bits)
SeqFlags	2	0	2	0	Segmentation Flag
SeqCount	2	2	14	0	Source Sequence Counter
PcktDataLen	4	0	16	0	Packet Data Length
PUSVersion	6	0	4	0	PUS Version
SpaceTimeRefStatus	6	4	4	0	Space Time Reference Status
ServType	7	0	8	0	PUS Service Type
ServSubType	8	0	8	0	PUS Service Sub Type
DestId	9	0	8	0	Destination Identifier
Time	10	0	6*8	0	CUC Time (6 bytes)
					Total bits: 128 Total bytes: 16.0 Total words: 8.0

4 Service Overview

Table 4.1 lists the services supported by the PUS Extension of the CORDET Framework and table 4.2 lists the commands and reports supported in each service.

Table 4.1: List of Supported Services

Type	Acron.	Name
1	Ver	Request Verification Service
3	Hk	Housekeeping Service
5	Evt	Event Reporting Service
11	Scd	Time Based Scheduling Service
12	Mon	On Board Monitoring Service
13	Lpt	Large Packet Transfer Service
17	Tst	Test Service
255	Dum	Dummy Service

Table 4.2: List of Supported Commands/Reports

Type	CORDET Name	PUS Name
TM(1,1)	SuccAccRep	Successful Acceptance Verification Report
TM(1,2)	FailedAccRep	Failed Acceptance Verification Report
TM(1,3)	SuccStartRep	Successful Start of Execution Verification Report
TM(1,4)	FailedStartRep	Failed Start of Execution Verification Report
TM(1,5)	SuccPrgrRep	Successful Progress of Execution Verification Report
TM(1,6)	FailedPrgrRep	Failed Progress of Execution Verification Report
TM(1,7)	SuccTermRep	Successful Completion of Execution Verification Report
TM(1,8)	FailedTermRep	Failed Completion of Execution Verification Report
TM(1,10)	FailedRoutingRep	Failed Routing Verification Report
TC(3,1)	CreHkCmd	Create a Housekeeping Parameter Report Structure
TC(3,2)	CreDiagCmd	Create a Diagnostic Parameter Report Structure
TC(3,3)	DelHkCmd	Delete a Housekeeping Parameter Report Structure
TC(3,4)	DelDiagCmd	Delete a Diagnostic Parameter Report Structure
TC(3,5)	EnbHkCmd	Enable Periodic Generation of a Housekeeping Parameter Report Structure
TC(3,6)	DisHkCmd	Disable Periodic Generation of a Housekeeping Parameter Report Structure
TC(3,7)	EnbDiagCmd	Enable Periodic Generation of a Diagnostic Parameter Report Structure
TC(3,8)	DisDiagCmd	Disable Periodic Generation of a Diagnostic Parameter Report Structure
TC(3,9)	RepStructHkCmd	Report Housekeeping Parameter Report Structure
TM(3,10)	RepStructHkRep	Housekeeping Parameter Report Structure Report
TC(3,11)	RepStructDiagCmd	Report Diagnostic Parameter Report Structure
TM(3,12)	RepStructDiagRep	Diagnostic Parameter Report Structure Report
TM(3,25)	Rep	Housekeeping Parameter Report
TM(3,26)	DiagRep	Diagnostic Parameter Report
TC(3,27)	OneShotHkCmd	Generate One-Shot Report for Housekeeping Parameters
TC(3,28)	OneShotDiagCmd	Generate One-Shot Report for Diagnostic Parameters

Type	CORDET Name	PUS Name
TC(3,31)	ModPerHkCmd	Modify Collection Interval of Housekeeping Report Structure
TC(3,32)	ModPerDiagCmd	Modify Collection Interval of Diagnostic Report Structure
TM(5,1)	Rep1	Informative Event Report (Level 1)
TM(5,2)	Rep2	Low Severity Event Report (Level 2)
TM(5,3)	Rep3	Medium Severity Event Report (Level 3)
TM(5,4)	Rep4	High Severity Event Report (Level 4)
TC(5,5)	EnbCmd	Enable Generation of Event Identifiers
TC(5,6)	DisCmd	Disable Generation of Event Identifiers
TC(5,7)	RepDisCmd	Report the List of Disabled Event Identifiers
TM(5,8)	DisRep	Disabled Event Identifier Report
TC(11,1)	EnbTbsCmd	Enable Time-Based Schedule Execution Function
TC(11,2)	DisTbsCmd	Disable Time-Based Schedule Execution Function
TC(11,3)	ResTbsCmd	Reset Time-Based Schedule
TC(11,4)	InsTbaCmd	Insert Activities into Time-Based Schedule
TC(11,5)	DelTbaCmd	Delete Activities from Time-Based Schedule
TC(11,20)	EnbSubSchedCmd	Enable Time-Based Sub-Schedules
TC(11,21)	DisSubSchedCmd	Disable Time-Based Sub-Schedules
TC(11,22)	CreGrpCmd	Create Time-Based Scheduling Groups
TC(11,23)	DelGrpCmd	Delete Time-Based Scheduling Groups
TC(11,24)	EnbGrpCmd	Enable Time-Based Scheduling Groups
TC(11,25)	DisGrpCmd	Disable Time-Based Scheduling Groups
TC(11,26)	RepGrpCmd	Report Status of Time-Based Scheduling Groups
TM(11,27)	GrpRep	Time-Based Scheduling Group Status Report
TC(12,1)	EnbParMonDefCmd	Enable Parameter Monitoring Definitions
TC(12,2)	DisParMonDefCmd	Disable Parameter Monitoring Definitions
TC(12,3)	ChgTransDelCmd	Change Maximum Transition Reporting Delay
TC(12,4)	DelAllParMonCmd	Delete All Parameter Monitoring Definitions
TC(12,5)	AddParMonDefCmd	Add Parameter Monitoring Definitions
TC(12,6)	DelParMonDefCmd	Delete Parameter Monitoring Definitions
TC(12,7)	ModParMonDefCmd	Modify Parameter Monitoring Definitions
TC(12,8)	RepParMonDefCmd	Report Parameter Monitoring Definitions
TM(12,9)	RepParMonDefRep	Parameter Monitoring Definition Report
TC(12,10)	RepOutOfLimitsCmd	Report Out Of Limit Monitors
TM(12,11)	RepOutOfLimitsRep	Out Of Limit Monitors Report
TM(12,12)	CheckTransRep	Check Transition Report
TC(12,13)	RepParMonStatCmd	Report Status of Parameter Monitors
TM(12,14)	RepParMonStatRep	Parameter Monitor Status Report

Type	CORDET Name	PUS Name
TC(12,15)	EnbParMonFuncCmd	Enable Parameter Monitoring Function
TC(12,16)	DisParMonFuncCmd	Disable Parameter Monitoring Function
TC(12,17)	EnbFuncMonCmd	Enable Functional Monitoring Function
TC(12,18)	DisFuncMonCmd	Disable Functional Monitoring Function
TC(12,19)	EnbFuncMonDefCmd	Enable Functional Monitoring Definitions
TC(12,20)	DisFuncMonDefCmd	Disable Functional Monitoring Definitions
TC(12,21)	ProtFuncMonDefCmd	Protect Functional Monitoring Definitions
TC(12,22)	UnprotFuncMonDefCmd	Unprotect Functional Monitoring Definitions
TC(12,23)	AddFuncMonDefCmd	Add Functional Monitoring Definitions
TC(12,24)	DelFuncMonDefCmd	Delete Functional Monitoring Definitions
TC(12,25)	RepFuncMonDefCmd	Report Functional Monitoring Definitions
TM(12,26)	RepFuncMonDefRep	Report Functional Monitoring Definitions
TC(12,27)	RepFuncMonStatCmd	Report Status of Functional Monitors
TM(12,28)	RepFuncMonStatRep	Status of Functional Monitors Report
TM(13,1)	DownFirstRep	First Downlink Part Report
TM(13,2)	DownInterRep	Intermediate Downlink Report
TM(13,3)	DownLastRep	Last Downlink Part Report
TC(13,9)	UpFirstCmd	First Uplink Part
TC(13,10)	UpInterCmd	Intermediate Uplink Part
TC(13,11)	UpLastCmd	Last Uplink Part
TM(13,16)	UpAbortRep	Large Packet Uplink Abortion Report
TC(13,129)	StartDownCmd	Trigger Large Packet Down-Transfer
TC(13,130)	AbortDownCmd	Abort Large Packet Down-Transfer
TC(17,1)	AreYouAliveCmd	Perform Are-You-Alive Connection Test
TM(17,2)	AreYouAliveRep	Are-You-Alive Connection Report
TC(17,3)	ConnectCmd	Perform On-Board Connection Test
TM(17,4)	ConnectRep	On-Board Connection Test Report
TC(255,1)	Sample1	Sample 1 Command

5 Detailed Definition of PUS Services

This section describes the content of all the commands and reports supported by the PUS Extension of the CORDET Framework Software. Each command or report is described in a dedicated table which lists the parameters in the command or report body (the structure of their headers is described in section 3).

5.1 TM(1,1) SuccAccRep

Report generated to mark the successful acceptance of an incoming command

Table 5.1: SuccAccRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of telecommand being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of telecommand being acknowledged
				Total bits: 32 Total bytes: 4.0 Total words: 2.0

5.2 TM(1,2) FailedAccRep

Report generated to mark the acceptance failure of an incoming command

Table 5.2: FailedAccRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of command being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of command being acknowledged
TcFailCode	4	0	8	Failure Identification Code
TcFailData	5	0	32	Failure data (interpretation depends on the value of the failure code, see description of FailCode)
TcType	9	0	8	Type of Acknowledged TC
TcSubType	10	0	8	Subtype of Acknowledged TC
TcDisc	11	0	16	Discriminant of Acknowledged TC
				Total bits: 104 Total bytes: 13.0 Total words: 6.5

5.3 TM(1,3) SuccStartRep

Report generated to mark the successful start of execution of an incoming command

Table 5.3: SuccStartRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of command being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of command being acknowledged
				Total bits: 32 Total bytes: 4.0 Total words: 2.0

5.4 TM(1,4) FailedStartRep

Report generated to mark the start of execution failure of an incoming command

Table 5.4: FailedStartRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet Version Number
TcPcktId	0	3	13	Packet Identifier of Acknowledged TC
TcPcktSeqCtrl	2	0	16	Packet Seq. Control of Acknowledged TC
TcFailCode	4	0	8	Failure Identification Code
TcFailData	5	0	32	Failure data (interpretation depends on the value of the failure code, see description of FailCode)
TcType	9	0	8	Type of Acknowledged TC
TcSubType	10	0	8	Subtype of Acknowledged TC
TcDisc	11	0	16	Discriminant of Acknowledged TC
				Total bits: 104 Total bytes: 13.0 Total words: 6.5

5.5 TM(1,5) SuccPrgrRep

Report generated to mark the successful completion of an execution step of an incoming command

Table 5.5: SuccPrgrRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of command being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of command being acknowledged
TcPrgStep	4	0	16	Identifier of the progress step which triggered the acknowledge report
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.6 TM(1,6) FailedPrgrRep

Report generated to mark the failure of an execution step of an incoming command

Table 5.6: FailedPrgrRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet Version Number
TcPcktId	0	3	13	Packet Identifier of Acknowledged TC
TcPcktSeqCtrl	2	0	16	Packet Seq. Control of Acknowledged TC
TcFailCode	4	0	8	Failure Identification Code
TcFailData	5	0	32	Failure data (interpretation depends on the value of the failure code, see description of FailCode)
TcType	9	0	8	Type of Acknowledged TC
TcSubType	10	0	8	Subtype of Acknowledged TC
TcDisc	11	0	16	Discriminant of Acknowledged TC
TcPrgStep	13	0	16	Progress step at which the failure was triggered
				Total bits: 120 Total bytes: 15.0 Total words: 7.5

5.7 TM(1,7) SuccTermRep

Report generated to mark the successful completion of execution of an incoming command

Table 5.7: SuccTermRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of command being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of command being acknowledged
				Total bits: 32 Total bytes: 4.0 Total words: 2.0

5.8 TM(1,8) FailedTermRep

Report generated to mark the failure to complete execution of an incoming command

Table 5.8: FailedTermRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet Version Number
TcPcktId	0	3	13	Packet Identifier of Acknowledged TC
TcPcktSeqCtrl	2	0	16	Packet Seq. Control of Acknowledged TC
TcFailCode	4	0	8	Failure Identification Code
TcFailData	5	0	32	Failure data (interpretation depends on the value of the failure code, see description of FailCode)
TcType	9	0	8	Type of Acknowledged TC
TcSubType	10	0	8	Subtype of Acknowledged TC
TcDisc	11	0	16	Discriminant of Acknowledged TC
				Total bits: 104 Total bytes: 13.0 Total words: 6.5

5.9 TM(1,10) FailedRoutingRep

Report generated to mark the failure to route an incoming command to its final destination

Table 5.9: FailedRoutingRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet Version Number
TcPcktId	0	3	13	Packet Identifier of Acknowledged TC
TcPcktSeqCtrl	2	0	16	Packet Seq. Control of Acknowledged TC
InvDest	4	0	8	invalid Destination for Rerouting Failure
TcType	5	0	8	Type of Acknowledged TC
TcSubType	6	0	8	Subtype of Acknowledged TC
TcDisc	7	0	16	Discriminant of Acknowledged TC
				Total bits: 72 Total bytes: 9.0 Total words: 4.5

5.10 TC(3,1) CreHkCmd

Create a housekeeping report structure

Table 5.10: CreHkCmd

Name	Byte	Bit	Size	Description
SID	0	0	16	The structure identifier (SID) of the packet to be created
CollectionInterval	2	0	16	Collection Interval
N1	4	0	8	The number of parameters in the housekeeping report to be created
- N1ParamId[1]	5	0	16	The identifiers of the simply commutated parameters in the report to be created
- ...				
- N1ParamId[N1]	-	-	16	The identifiers of the simply commutated parameters in the report to be created
NFA	-	-	8	The number of super-commutated groups of parameters
- SCSampleRepNum[1]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[1]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[1][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[1][N2]	-	-	16	Parameter ID
- ...				
- SCSampleRepNum[NFA]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[NFA]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[NFA][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[NFA][N2]	-	-	16	Parameter ID

5.11 TC(3,2) CreDiagCmd

Create a diagnostic report structure

Table 5.11: CreDiagCmd

Name	Byte	Bit	Size	Description
SID	0	0	16	The structure identifier (SID) of the packet to be created
CollectionInterval	2	0	16	Collection Interval
N1	4	0	8	The number of parameters in the diagnostic report to be created
- N1ParamId[1]	5	0	16	The identifiers of the simply commutated parameters in the report to be created
- ...				
- N1ParamId[N1]	-	-	16	The identifiers of the simply commutated parameters in the report to be created
NFA	-	-	8	The number of super-commutated groups of parameters
- SCSampleRepNum[1]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[1]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[1][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[1][N2]	-	-	16	Parameter ID
- ...				
- SCSampleRepNum[NFA]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[NFA]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[NFA][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[NFA][N2]	-	-	16	Parameter ID

5.12 TC(3,3) DelHkCmd

Delete one or more housekeeping report definitions

Table 5.12: DelHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	The number of report definitions to be deleted
- SID[1]	1	0	16	The structure identifiers (SIDs) of the report definitions to be deleted
- ...				
- SID[N]	-	-	16	The structure identifiers (SIDs) of the report definitions to be deleted

5.13 TC(3,4) DelDiagCmd

Delete one or more diagnostic report definitions

Table 5.13: DelDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	The number of report definitions to be deleted
- SID[1]	1	0	16	The structure identifier (SID) of a report to be deleted
- ...				
- SID[N]	-	-	16	The structure identifier (SID) of a report to be deleted

5.14 TC(3,5) EnbHkCmd

Enable the periodic generation of one or more housekeeping report structures

Table 5.14: EnbHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be enabled
- SID[1]	1	0	16	SID to be enabled
- ...				
- SID[N]	-	-	16	SID to be enabled

5.15 TC(3,6) DisHkCmd

Disable the periodic generation of one or more housekeeping report structures

Table 5.15: DisHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be disabled
- SID[1]	1	0	16	SID to be disabled
- ...				
- SID[N]	-	-	16	SID to be disabled

5.16 TC(3,7) EnbDiagCmd

Enable the periodic generation of one or more diagnostic report structures

Table 5.16: EnbDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be enabled
- SID[1]	1	0	16	SID to be enabled
- ...				
- SID[N]	-	-	16	SID to be enabled

5.17 TC(3,8) DisDiagCmd

Disable the periodic generation of one or more diagnostic report structures

Table 5.17: DisDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be disabled
- SID[1]	1	0	16	SID to be disabled
- ...				
- SID[N]	-	-	16	SID to be disabled

5.18 TC(3,9) RepStructHkCmd

This command carries a list of SIDs. For each SID, it triggers the generation of a (3,10) report with the definition of the housekeeping report structure for that SID.

Table 5.18: RepStructHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be reported
- SID[1]	1	0	16	SID to be reported
- ...				
- SID[N]	-	-	16	SID to be reported

5.19 TM(3,10) RepStructHkRep

Report carrying the definition of a housekeeping report structure generated in response to a (3,9) command.

Table 5.19: RepStructHkRep

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
PerGenActionStatus	2	0	8	Flag indicating whether periodic generation of the packet is enabled or disabled
CollectionInterval	3	0	16	Collection Interval
N1	5	0	8	The number of simply commutated parameters
- N1ParamId[1]	6	0	16	Identifier of a simply commutated parameter
- ...				
- N1ParamId[N1]	-	-	16	Identifier of a simply commutated parameter
NFA	-	-	8	The number of super-commutated groups of parameters
- SCSampleRepNum[1]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[1]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[1][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[1][N2]	-	-	16	Parameter ID
- ...				
- SCSampleRepNum[NFA]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[NFA]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[NFA][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[NFA][N2]	-	-	16	Parameter ID

5.20 TC(3,11) RepStructDiagCmd

This command carries a list of SIDs. For each SID, it triggers the generation of a (3,12) report with the definition of the diagnostic report structure for that SID.

Table 5.20: RepStructDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be reported
- SID[1]	1	0	16	SID to be reported
- ...				
- SID[N]	-	-	16	SID to be reported

5.21 TM(3,12) RepStructDiagRep

Report carrying the definition of a diagnostic report structure generated in response to a (3,11) command.

Table 5.21: RepStructDiagRep

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
PerGenActionStatus	2	0	8	Flag indicating whether periodic generation of the packet is enabled or disabled
CollectionInterval	3	0	16	Collection Interval
N1	5	0	8	The number of simply commutated parameters
- N1ParamId[1]	6	0	16	Identifier of a simply commutated parameter
- ...				
- N1ParamId[N1]	-	-	16	Identifier of a simply commutated parameter
NFA	-	-	8	The number of super-commutated groups of parameters
- SCSampleRepNum[1]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[1]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[1][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[1][N2]	-	-	16	Parameter ID
- ...				
- SCSampleRepNum[NFA]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[NFA]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[NFA][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[NFA][N2]	-	-	16	Parameter ID

5.22 TM(3,25) Rep (SID_HK_CNT)

Housekeeping packet holding the cycle counters of the HK packets

Table 5.22: Rep (SID_HK_CNT)

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
cycleCnt	2	0	4*16	Cycle Counter for Reports in RDL
				Total bits: 80 Total bytes: 10.0 Total words: 5.0

5.23 TM(3,25) Rep (SID_N_OF_EVT)

Housekeeping packet holding number of generated events of each severity level

Table 5.23: Rep (SID_N_OF_EVT)

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
nOfDetectedEvs	2	0	4*16	Number of detected occurrences of events (one element for each severity level)
nOfDisabledEid	10	0	4*16	Number of disabled event identifiers (one element for each severity level)
				Total bits: 144 Total bytes: 18.0 Total words: 9.0

5.24 TM(3,26) DiagRep

Periodic Diagnostic Report (3,26)

Table 5.24: DiagRep

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.25 TC(3,27) OneShotHkCmd

Command (3,27) to generate a one-shot housekeeping report

Table 5.25: OneShotHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be generated in one-shot mode
- SID[1]	1	0	16	SID to be generated in one-shot mode
- ...				
- SID[N]	-	-	16	SID to be generated in one-shot mode

5.26 TC(3,28) OneShotDiagCmd

Command (3,28) to generate a one-shot diagnostic report

Table 5.26: OneShotDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be generated in one-shot mode
- SID[1]	1	0	16	SID to be generated in one-shot mode
- ...				
- SID[N]	-	-	16	SID to be generated in one-shot mode

5.27 TC(3,31) ModPerHkCmd

Command (3,31) to modify the collection period of a housekeeping report

Table 5.27: ModPerHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs whose collection interval is to be modified
- SID[1]	1	0	16	SID whose collection interval is to be modified
- CollectionInterval[1]	3	0	16	New collection interval
- ...				
- SID[N]	-	-	16	SID whose collection interval is to be modified
- CollectionInterval[N]	-	-	16	New collection interval

5.28 TC(3,32) ModPerDiagCmd

Command (3,31) to modify the collection period of a diagnostic report

Table 5.28: ModPerDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs whose collection interval is to be modified
- SID[1]	1	0	16	SID whose collection interval is to be modified
- CollectionInterval[1]	3	0	16	New collection interval
- ...				
- SID[N]	-	-	16	SID whose collection interval is to be modified
- CollectionInterval[N]	-	-	16	New collection interval

5.29 TM(5,1) Rep1 (EVT_DOWN_ABORT)

Triggered by the aborting of a down-transfer

Table 5.29: Rep1 (EVT_DOWN_ABORT)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
LptSmId	2	0	8	Identifier of LPT State Machine where the down-transfer abort occurred
				Total bits: 24 Total bytes: 3.0 Total words: 1.5

5.30 TM(5,1) Rep1 (EVT_DUMMY_1)

Dummy event with one dummy parameter used for testing

Table 5.30: Rep1 (EVT_DUMMY_1)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
Par	2	0	8	Dummy parameter for dummy event used for testing
				Total bits: 24 Total bytes: 3.0 Total words: 1.5

5.31 TM(5,1) Rep1 (EVT_UP_ABORT)

Triggered by the aborting of a up-transfer

Table 5.31: Rep1 (EVT_UP_ABORT)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
LptSmId	2	0	8	Identifier of LPT State Machine where the up-transfer abort occurred
				Total bits: 24 Total bytes: 3.0 Total words: 1.5

5.32 TM(5,2) Rep2 (EVT_CLST_FULL)

Generated when the Monitoring Function Procedure tries to add an entry to the Check Transition List but the list is full

Table 5.32: Rep2 (EVT_CLST_FULL)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.33 TM(5,2) Rep2 (EVT_DUMMY_2)

Dummy level 2 event (no parameters)

Table 5.33: Rep2 (EVT_DUMMY_2)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.34 TM(5,3) Rep3 (EVT_DUMMY_3)

Dummy level 3 event (one dummy parameter)

Table 5.34: Rep3 (EVT_DUMMY_3)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
Par	2	0	8	Dummy parameter for dummy level 3 event
				Total bits: 24 Total bytes: 3.0 Total words: 1.5

5.35 TM(5,3) Rep3 (EVT_FMON_FAIL)

Generated when a functional monitor has declared a failure

Table 5.35: Rep3 (EVT_FMON_FAIL)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.36 TM(5,3) Rep3 (EVT_MON_DEL_I)

Generated when a Delta Check Monitoring Procedure has detected an invalid parameter value of integer type

Table 5.36: Rep3 (EVT_MON_DEL_I)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
ParMonId	2	0	16	The identifier of the parameter monitor which reported the violation
MonParId	4	0	16	The identifier of the monitored parameter which went out of limits
ParMonCheckStatus	6	0	8	Last status of parameter monitor which reported limit violation
ParValueInt	7	0	32	Last value of monitored parameter which went out-of-limits
				Total bits: 88 Total bytes: 11.0 Total words: 5.5

5.37 TM(5,3) Rep3 (EVT_MON_DEL_R)

Generated when a Delta Check Monitoring Procedure has detected an invalid parameter value of real type

Table 5.37: Rep3 (EVT_MON_DEL_R)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
ParMonId	2	0	16	The identifier of the parameter monitor which reported the violation
MonParId	4	0	16	The identifier of the monitored parameter which went out of limits
ParMonCheckStatus	6	0	8	Last status of parameter monitor which reported limit violation
ParValueReal	7	0	32	Last value of monitored parameter which went out-of-limits
				Total bits: 88 Total bytes: 11.0 Total words: 5.5

5.38 TM(5,3) Rep3 (EVT_MON_LIM_I)

Generated when a Limit Check Monitoring Procedure has detected an invalid parameter value of integer type

Table 5.38: Rep3 (EVT_MON_LIM_I)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
ParMonId	2	0	16	The identifier of the parameter monitor which reported the violation
MonParId	4	0	16	The identifier of the monitored parameter which went out of limits
ParMonCheckStatus	6	0	8	Last status of parameter monitor which reported limit violation
ParValueInt	7	0	32	Last value of monitored parameter which went out-of-limits
				Total bits: 88 Total bytes: 11.0 Total words: 5.5

5.39 TM(5,3) Rep3 (EVT_MON_LIM_R)

Generated when a Limit Check Monitoring Procedure has detected an invalid parameter value of real type

Table 5.39: Rep3 (EVT_MON_LIM_R)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
ParMonId	2	0	16	The identifier of the parameter monitor which reported the violation
MonParId	4	0	16	The identifier of the monitored parameter which went out of limits
ParMonCheckStatus	6	0	8	Last status of parameter monitor which reported limit violation
ParValueReal	7	0	32	Last value of monitored parameter which went out-of-limits
				Total bits: 88 Total bytes: 11.0 Total words: 5.5

5.40 TM(5,4) Rep4 (EVT_DUMMY_4)

Dummy level 4 event (no parameters)

Table 5.40: Rep4 (EVT_DUMMY_4)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.41 TC(5,5) EnbCmd

Command to enable generation of a list of event identifiers

Table 5.41: EnbCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of event identifiers to be enabled
- EventId[1]	2	0	16	Event identifier to be enabled
- ...				
- EventId[N]	-	-	16	Event identifier to be enabled

5.42 TC(5,6) DisCmd

Command to disable generation of a list of event identifiers

Table 5.42: DisCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of event identifiers to be disabled
- EventId[1]	2	0	16	Event identifier to be disabled
- ...				
- EventId[N]	-	-	16	Event identifier to be disabled

5.43 TC(5,7) RepDisCmd

This command triggers the generation of a (5,8) report holding the list of disabled event identifiers

This packet does not have any parameters.

5.44 TM(5,8) DisRep

Report generated in response to a (5,7) command carrying the list of disabled Event Identifiers

Table 5.43: DisRep

Name	Byte	Bit	Size	Description
N	0	0	16	The number of disabled event identifiers
- EventId[1]	2	0	16	Event Identifier
- EventId[1]	4	0	16	Event Identifier
- ...				
- EventId[N]	-	-	16	Event Identifier
- EventId[N]	-	-	16	Event Identifier

5.45 TC(11,1) EnbTbsCmd

Command to enable the time-based schedule execution function

This packet does not have any parameters.

5.46 TC(11,2) DisTbsCmd

Command to disable the time-based schedule execution function

This packet does not have any parameters.

5.47 TC(11,3) ResTbsCmd

Command to reset the time-based schedule

This packet does not have any parameters.

5.48 TC(11,4) InsTbaCmd

Command to insert one or more time-based activities (TBAs) into the time-based schedule (TBS)

Table 5.44: InsTbaCmd

Name	Byte	Bit	Size	Description
SubSchedId	0	0	8	Sub-schedule to which the activities are assigned
N	1	0	16	Number of activities to be inserted in the time-based schedule
- GroupId[1]	3	0	8	Schedule group to which this activity is assigned
- RelTime[1]	4	0	48	Release time of activity
- Request[1]	10	0	undefined	Command which implements the activity
- ...				
- GroupId[N]	-	-	8	Schedule group to which this activity is assigned
- RelTime[N]	-	-	48	Release time of activity
- Request[N]	-	-	undefined	Command which implements the activity

5.49 TC(11,5) DelTbaCmd

Command to delete one or more time-based activities (TBAs) from the time-based schedule (TBS)

Table 5.45: DelTbaCmd

Name	Byte	Bit	Size	Description
N	0	0	16	Number of activities to be inserted in the time-based schedule
- AppId[1]	2	0	8	Source of command embedded in activity to be deleted
- APID[1]	3	0	11	APID of command embedded in activity to be deleted
- SrcSeqCnt[1]	4	3	14	Source sequence count of command embedded in activity to be deleted
- ...				
- AppId[N]	-	-	8	Source of command embedded in activity to be deleted
- APID[N]	-	-	11	APID of command embedded in activity to be deleted
- SrcSeqCnt[N]	-	-	14	Source sequence count of command embedded in activity to be deleted

5.50 TC(11,20) EnbSubSchedCmd

Command to enable one or more time-based sub-schedules

Table 5.46: EnbSubSchedCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of sub-schedule identifiers to be enabled
- SubSchedId[1]	2	0	8	The identifier of a sub-schedule to be enabled
- ...				
- SubSchedId[N]	-	-	8	The identifier of a sub-schedule to be enabled

5.51 TC(11,21) DisSubSchedCmd

Command to disable one or more time-based sub-schedules

Table 5.47: DisSubSchedCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of sub-schedule identifiers to be disabled
- SubSchedId[1]	2	0	8	The identifier of a sub-schedule to be disabled
- ...				
- SubSchedId[N]	-	-	8	The identifier of a sub-schedule to be disabled

5.52 TC(11,22) CreGrpCmd

Command to create one or more scheduling groups

Table 5.48: CreGrpCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of groups to be created
- GroupId[1]	2	0	8	The identifier of a group to be created
- isGroupEnabled[1]	3	0	8	The initial enable status of the group
- ...				
- GroupId[N]	-	-	8	The identifier of a group to be created
- isGroupEnabled[N]	-	-	8	The initial enable status of the group

5.53 TC(11,23) DelGrpCmd

Command to delete one or more scheduling groups

Table 5.49: DelGrpCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of groups to be deleted
- GroupId[1]	2	0	8	The identifier of a group to be deleted
- ...				
- GroupId[N]	-	-	8	The identifier of a group to be deleted

5.54 TC(11,24) EnbGrpCmd

Command to enable one or more scheduling groups

Table 5.50: EnbGrpCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of groups to be enabled (if this is zero, then all groups are enabled)
- GroupId[1]	2	0	8	The identifier of a group to be enabled
- ...				
- GroupId[N]	-	-	8	The identifier of a group to be enabled

5.55 TC(11,25) DisGrpCmd

Command to disable one or more scheduling groups

Table 5.51: DisGrpCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of groups to be enabled (if this is zero, then all groups currently in use are disabled)
- GroupId[1]	2	0	8	The identifier of a group to be enabled
- ...				
- GroupId[N]	-	-	8	The identifier of a group to be enabled

5.56 TC(11,26) RepGrpCmd

Command to trigger the generation of a (11,27) report carrying the status of the scheduling groups

This packet does not have any parameters.

5.57 TM(11,27) GrpRep

Report generated in response to a (11,26) command to report the status of the scheduling groups

Table 5.52: GrpRep

Name	Byte	Bit	Size	Description
N	0	0	16	Number of groups being reported
- GroupId[1]	2	0	8	The identifier of a group being reported
- isGroupEnabled[1]	3	0	8	The enable status of a group being reported
- ...				
- GroupId[N]	-	-	8	The identifier of a group being reported
- isGroupEnabled[N]	-	-	8	The enable status of a group being reported

5.58 TC(12,1) EnbParMonDefCmd

Command to enable one or more monitoring definitions

Table 5.53: EnbParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of monitoring definitions to be enabled.
- ParMonId[1]	1	0	16	Identifier of Parameter Monitor
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of Parameter Monitor

5.59 TC(12,2) DisParMonDefCmd

Command to disable one or more monitoring definitions

Table 5.54: DisParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of monitoring definitions to be disabled,
- ParMonId[1]	1	0	16	Identifier of Parameter Monitor
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of Parameter Monitor

5.60 TC(12,3) ChgTransDelCmd

Command to change the maximum delay after which the content of the check transition list (CTL) is reported through a (12,12) report

Table 5.55: ChgTransDelCmd

Name	Byte	Bit	Size	Description
maxRepDelay	0	0	16	Maximum reporting delay
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.61 TC(12,4) DelAllParMonCmd

Command to delete all parameter monitoring definitions

This packet does not have any parameters.

5.62 TC(12,5) AddParMonDefCmd

Command to add one or more parameter definitions

Table 5.56: AddParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of parameter definitions
- ParMonId[1]	1	0	16	Identifier of parameter monitor to be added by telecommand
- MonParId[1]	3	0	16	Identifier of Monitored Parameter
- ValCheckParId[1]	5	0	16	Identifier of Validity Parameter
- ExpValCheckMask[1]	7	0	16	Expected Value Check Mask
- ValCheckExpVal[1]	9	0	undefined	Expected Value for Validity Check
- CheckTypeData[1]	-	-	undefined	For expected value check, this parameter consists of: mask, expected value and event identifier. For limit checks, it consists of: low limit, event identifier for low limit, high limit, event identifier for high limit. For delta checks, it consists of: low delta threshold, event identifier for low threshold, high delta threshold, event identifiers for high thresholds. In all cases, an event identifier of zero indicates that no event is associated to the violation.
- RepNmb[1]	-	-	8	Repetition Number
- CheckType[1]	-	-	8	Type of Monitor Procedure
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of parameter monitor to be added by telecommand
- MonParId[NParMon]	-	-	16	Identifier of Monitored Parameter
- ValCheckParId[NParMon]	-	-	16	Identifier of Validity Parameter
- ExpValCheckMask[NParMon]	-	-	16	Expected Value Check Mask
- ValCheckExpVal[NParMon]	-	-	undefined	Expected Value for Validity Check
- CheckTypeData[NParMon]	-	-	undefined	For expected value check, this parameter consists of: mask, expected value and event identifier. For limit checks, it consists of: low limit, event identifier for low limit, high limit, event identifier for high limit. For delta checks, it consists of: low delta threshold, event identifier for low threshold, high delta threshold, event identifiers for high thresholds. In all cases, an event identifier of zero indicates that no event is associated to the violation.

Name	Byte	Bit	Size	Description
- RepNmb[NParMon]	-	-	8	Repetition Number
- CheckType[NParMon]	-	-	8	Type of Monitor Procedure
CheckTypeData	-	-	undefined	For expected value check, this parameter consists of: mask, expected value and event identifier. For limit checks, it consists of: low limit, event identifier for low limit, high limit, event identifier for high limit. For delta checks, it consists of: low delta threshold, event identifier for low threshold, high delta threshold, event identifiers for high thresholds. In all cases, an event identifier of zero indicates that no event is associated to the violation.

5.63 TC(12,6) DelParMonDefCmd

Command to delete one or more parameter monitoring definitions

Table 5.57: DelParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of parameter monitoring definitions to be deleted
- ParMonId[1]	1	0	16	The identifier of the parameter monitoring definition to be deleted
- ...				
- ParMonId[NParMon]	-	-	16	The identifier of the parameter monitoring definition to be deleted

5.64 TC(12,7) ModParMonDefCmd

Command to modify one or more parameter definitions

Table 5.58: ModParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of parameter monitoring definitions to be modified
MonParId	1	0	16	The identifier of the parameter monitor to be modified
MonParId	3	0	16	The identifier of the data pool item to be monitored by the modified parameter monitor
RepNmb	5	0	8	The repetition number of the modified parameter monitor
CheckType	6	0	8	The monitoring check type of the modified parameter monitor
CheckTypeData	7	0	undefined	For expected value check, this parameter consists of: mask, expected value and event identifier. For limit checks, it consists of: low limit, event identifier for low limit, high limit, event identifier for high limit. For delta checks, it consists of: low delta threshold, event identifier for low threshold, high delta threshold, event identifiers for high thresholds. In all cases, an event identifier of zero indicates that no event is associated to the violation.

5.65 TC(12,8) RepParMonDefCmd

This command triggers the generation of a (12,9) report carrying one or more parameter monitor definitions

Table 5.59: RepParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	The number of parameter monitoring definitions to be reported or zero if all parameter monitoring definitions must be reported
- ParMonId[1]	1	0	16	Identifier of a parameter monitor to be reported
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of a parameter monitor to be reported

5.66 TM(12,9) RepParMonDefRep

Report generated in response to a (12,8) command to report one or more monitoring definitions.

This packet does not have any parameters.

5.67 TC(12,10) RepOutOfLimitsCmd

This command triggers the generation of a (12,11) report holding the parameter monitors which are out of limits

This packet does not have any parameters.

5.68 TM(12,11) RepOutOfLimitsRep

Report generated in response to a (12,10) command carrying the parameter monitors which are out of limits

Table 5.60: RepOutOfLimitsRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of out of limit parameter monitors which are reported
- MonParId[1]	1	0	16	Identifier of data pool item monitored by parameter monitor
- ParMonId[1]	3	0	16	Identifier of Parameter Monitor
- CheckType[1]	5	0	8	Type of Monitor Procedure
- ParValue[1]	6	0	undefined	Parameter Value at Monitoring Violation
- LimitCrossed[1]	-	-	undefined	Limit Crossed at Monitoring Violation
- ParMonCheckStatus[1]	-	-	8	Parameter Monitor Checking Status
- ParMonPrevCheckStatus[1]	-	-	8	Checking Status Before Monitoring Violation
- TransTime[1]	-	-	48	Time of Monitoring Violation Transition
- ...				
- MonParId[NParMon]	-	-	16	Identifier of data pool item monitored by parameter monitor
- ParMonId[NParMon]	-	-	16	Identifier of Parameter Monitor
- CheckType[NParMon]	-	-	8	Type of Monitor Procedure
- ParValue[NParMon]	-	-	undefined	Parameter Value at Monitoring Violation
- LimitCrossed[NParMon]	-	-	undefined	Limit Crossed at Monitoring Violation
- ParMonCheckStatus[NParMon]	-	-	8	Parameter Monitor Checking Status
- ParMonPrevCheckStatus[NParMon]	-	-	8	Checking Status Before Monitoring Violation
- TransTime[NParMon]	-	-	48	Time of Monitoring Violation Transition

5.69 TM(12,12) CheckTransRep

Report carrying the content of the Check Transition List (CTL).

Table 5.61: CheckTransRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of check transitions in the report
- ParMonId[1]	1	0	16	Identifier of Parameter Monitor
- MonParId[1]	3	0	16	Identifier of Monitored Parameter
- CheckType[1]	5	0	8	Type of Monitor Procedure
- ExpValCheckMask[1]	6	0	16	Expected Value Check Mask
- ParValue[1]	8	0	undefined	Parameter Value at Monitoring Violation
- LimitCrossed[1]	-	-	undefined	Limit Crossed at Monitoring Violation
- ParMonPrevCheckStatus[1]	-	-	8	Checking Status Before Monitoring Violation
- ParMonCheckStatus[1]	-	-	8	Parameter Monitor Checking Status
- TransTime[1]	-	-	48	Time of Monitoring Violation Transition
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of Parameter Monitor
- MonParId[NParMon]	-	-	16	Identifier of Monitored Parameter
- CheckType[NParMon]	-	-	8	Type of Monitor Procedure
- ExpValCheckMask[NParMon]	-	-	16	Expected Value Check Mask
- ParValue[NParMon]	-	-	undefined	Parameter Value at Monitoring Violation
- LimitCrossed[NParMon]	-	-	undefined	Limit Crossed at Monitoring Violation
- ParMonPrevCheckStatus[NParMon]	-	-	8	Checking Status Before Monitoring Violation
- ParMonCheckStatus[NParMon]	-	-	8	Parameter Monitor Checking Status
- TransTime[NParMon]	-	-	48	Time of Monitoring Violation Transition

5.70 TC(12,13) RepParMonStatCmd

This command triggers the generation of a (12,14) report carrying the status of all parameter monitors

This packet does not have any parameters.

5.71 TM(12,14) RepParMonStatRep

Report generated in response to a (12,13) report carrying the status of all currently defined parameter monitors

Table 5.62: RepParMonStatRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of parameter monitors whose status is reported by the telecommand
- ParMonId[1]	1	0	16	Identifier of a parameter monitor
- ParMonCheckStatus[1]	3	0	8	Current checking status of the parameter monitor
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of a parameter monitor
- ParMonCheckStatus[NParMon]	-	-	8	Current checking status of the parameter monitor

5.72 TC(12,15) EnbParMonFuncCmd

Command to enable the monitoring function

This packet does not have any parameters.

5.73 TC(12,16) DisParMonFuncCmd

Command to disable the parameter monitoring function

This packet does not have any parameters.

5.74 TC(12,17) EnbFuncMonCmd

Command to enable the functional monitoring function

This packet does not have any parameters.

5.75 TC(12,18) DisFuncMonCmd

Command to disable the functional monitoring function

This packet does not have any parameters.

5.76 TC(12,19) EnbFuncMonDefCmd

Command to enable one or more functional monitoring definitions

Table 5.63: EnbFuncMonDefCmd

Name	Byte	Bit	Size	Description
NFuncMon	0	0	8	The number of functional monitoring definitions to be enabled
- FuncMonId[1]	1	0	8	Identifier of a functional monitor to be enabled
- ...				
- FuncMonId[NFuncMon]	-	-	8	Identifier of a functional monitor to be enabled

5.77 TC(12,20) DisFuncMonDefCmd

Command to disable one ore more functional monitoring definitions

Table 5.64: DisFuncMonDefCmd

Name	Byte	Bit	Size	Description
NFuncMon	0	0	8	The number of functional monitors to be disabled
- FuncMonId[1]	1	0	8	Identifer of functional monitor to be disabled
- ...				
- FuncMonId[NFuncMon]	-	-	8	Identifer of functional monitor to be disabled

5.78 TC(12,21) ProtFuncMonDefCmd

Command to protect one or more functional monitoring definitions

Table 5.65: ProtFuncMonDefCmd

Name	Byte	Bit	Size	Description
NFuncMon	0	0	8	The number of functional monitors to be protected
- FuncMonId[1]	1	0	8	Identifier of functional monitor to be protected
- ...				
- FuncMonId[NFuncMon]	-	-	8	Identifier of functional monitor to be protected

5.79 TC(12,22) UnprotFuncMonDefCmd

Command to unprotect one or more functional monitoring definitions

Table 5.66: UnprotFuncMonDefCmd

Name	Byte	Bit	Size	Description
FuncMonId	0	0	8	The number of functional monitor to be unprotected
- FuncMonId[1]	1	0	8	Identifier of a functional monitor to be unprotected
- ...				
- FuncMonId[FuncMonId]	-	-	8	Identifier of a functional monitor to be unprotected

5.80 TC(12,23) AddFuncMonDefCmd

Command to add one or more functional monitoring definitions

Table 5.67: AddFuncMonDefCmd

Name	Byte	Bit	Size	Description
FuncMonId	0	0	8	Number of functional monitor definitions to be added
- FuncMonId[1]	1	0	8	Identifier of functional monitor to be added
- ValCheckParId[1]	2	0	16	Identifier of Validity Parameter
- ValCheckParMask[1]	4	0	undefined	Mask for Validity Check
- ValCheckExpVal[1]	-	-	undefined	Expected Value for Validity Check
- EvtId[1]	-	-	16	Event Identifier
- MinFailNmb[1]	-	-	8	Minimum Failing Number
- NFuncMon[1]	-	-	8	Number of parameter monitors attached to functional monitor
- ParMonId[1]	-	-	16	Identifier of Parameter Monitor
- ...				
- FuncMonId[FuncMonId]	-	-	8	Identifier of functional monitor to be added
- ValCheckParId[FuncMonId]	-	-	16	Identifier of Validity Parameter
- ValCheckParMask[FuncMonId]	-	-	undefined	Mask for Validity Check
- ValCheckExpVal[FuncMonId]	-	-	undefined	Expected Value for Validity Check
- EvtId[FuncMonId]	-	-	16	Event Identifier
- MinFailNmb[FuncMonId]	-	-	8	Minimum Failing Number
- NFuncMon[FuncMonId]	-	-	8	Number of parameter monitors attached to functional monitor
- ParMonId[FuncMonId]	-	-	16	Identifier of Parameter Monitor

5.81 TC(12,24) DelFuncMonDefCmd

Command to delete one or more functional monitoring definitions to the FMDL

Table 5.68: DelFuncMonDefCmd

Name	Byte	Bit	Size	Description
NFuncMon	0	0	8	Number of functional monitors to be deleted
- FuncMonId[1]	1	0	8	Identifier of functional monitor to be deleted
- ...				
- FuncMonId[NFuncMon]	-	-	8	Identifier of functional monitor to be deleted

5.82 TC(12,25) RepFuncMonDefCmd

This command triggers the generation of a (12,26) report carrying the definition of one or more functional monitors

Table 5.69: RepFuncMonDefCmd

Name	Byte	Bit	Size	Description
FuncMonId	0	0	8	Number of functional monitors whose definition is to be reported
- FuncMonId[1]	1	0	8	Identifier of functional monitor whose definition is to be reported
- ...				
- FuncMonId[FuncMonId]	-	-	8	Identifier of functional monitor whose definition is to be reported

5.83 TM(12,26) RepFuncMonDefRep

Report generated in response to a (12,25) command to carry the definition of some or all functional monitoring definitions

Table 5.70: RepFuncMonDefRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	The number of functional monitoring definitions in the report
- FuncMonId[1]	1	0	8	Identifier of Functional Monitor
- ValCheckParId[1]	2	0	16	Identifier of Validity Parameter
- ValCheckParMask[1]	4	0	undefined	Mask for Validity Check
- ValCheckParMask[1]	-	-	undefined	Mask for Validity Check
- ProtStatus[1]	-	-	8	Functional Monitor Protection Status
- FuncMonCheckStatus[1]	-	-	8	Functional Monitor Checking Status
- EvtId[1]	-	-	16	Event Identifier
- MinFailNmb[1]	-	-	8	Minimum Failing Number
- NFuncMon[1]	-	-	8	The number of parameter monitors associated to the functional monitor
- - ParMonId[1][1]	-	-	16	Identifier of Parameter Monitor
- - ...				
- - ParMonId[1][NFuncMon]	-	-	16	Identifier of Parameter Monitor
- ...				
- FuncMonId[NParMon]	-	-	8	Identifier of Functional Monitor
- ValCheckParId[NParMon]	-	-	16	Identifier of Validity Parameter
- ValCheckParMask[NParMon]	-	-	undefined	Mask for Validity Check
- ValCheckParMask[NParMon]	-	-	undefined	Mask for Validity Check
- ProtStatus[NParMon]	-	-	8	Functional Monitor Protection Status
- FuncMonCheckStatus[NParMon]	-	-	8	Functional Monitor Checking Status
- EvtId[NParMon]	-	-	16	Event Identifier
- MinFailNmb[NParMon]	-	-	8	Minimum Failing Number
- NFuncMon[NParMon]	-	-	8	The number of parameter monitors associated to the functional monitor

Name	Byte	Bit	Size	Description
- - ParMonId[NParMon][1]	-	-	16	Identifier of Parameter Monitor
- - ...				
- - ParMonId[NParMon][NFuncMon]	-	-	16	Identifier of Parameter Monitor

5.84 TC(12,27) RepFuncMonStatCmd

This command triggers the generation of a (12,28) report carrying the status of all functional monitors

This packet does not have any parameters.

5.85
TM(12,28)
RepFuncMonStatRep

Report generated in response to a (12,27) command carrying the status of all currently defined functional monitors

Table 5.71: RepFuncMonStatRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	The number of functional monitor statuses in the report
- FuncMonId[1]	1	0	8	Identifier of Functional Monitor
- ProtStatus[1]	2	0	8	Functional Monitor Protection Status
- IsEnabled[1]	3	0	8	Enable Status
- FuncMonCheckStatus[1]	4	0	8	Functional Monitor Checking Status
- ...				
- FuncMonId[NParMon]	-	-	8	Identifier of Functional Monitor
- ProtStatus[NParMon]	-	-	8	Functional Monitor Protection Status
- IsEnabled[NParMon]	-	-	8	Enable Status
- FuncMonCheckStatus[NParMon]	-	-	8	Functional Monitor Checking Status

5.86 TM(13,1) DownFirstRep

Report carrying the first part of a down-transfer

Table 5.72: DownFirstRep

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Down-transfer data (repetition value to be set to fill the packet)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.87 TM(13,2) DownInterRep

Report carrying an intermediate part of a down-transfer

Table 5.73: DownInterRep

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Down-transfer data (repetition value to be set to fill the packet)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.88 TM(13,3) DownLastRep

Report carrying the last part of a down-transfer

Table 5.74: DownLastRep

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Down-transfer data (repetition value is set dynamically by the host application based on the amount of data to be down-transferred)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.89 TC(13,9) UpFirstCmd

Command to carry the first part of an up-transfer

Table 5.75: UpFirstCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Up-transfer data (repetition value to be set to fill the packet)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.90 TC(13,10) UpInterCmd

Command to carry an intermediate part of an up-transfer

Table 5.76: UpInterCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Up-transfer data (repetition value to be set to fill the packet)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.91 TC(13,11) UpLastCmd

Command to carry the last part of an up-transfer

Table 5.77: UpLastCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Up-transfer data (repetition value is set dynamically)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.92 TM(13,16) UpAbortRep

Report to notify the abortion of an up-transfer

Table 5.78: UpAbortRep

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
FailReason	2	0	16	Transfer Failure Reason
				Total bits: 32 Total bytes: 4.0 Total words: 2.0

5.93 TC(13,129) StartDownCmd

Command to start a down-transfer

Table 5.79: StartDownCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.94 TC(13,130) AbortDownCmd

Command to abort a down-transfer

Table 5.80: AbortDownCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.95 TC(17,1) AreYouAliveCmd

Command to perform and Are-You-Alive Connection Test

This packet does not have any parameters.

5.96 TM(17,2) AreYouAliveRep

Report generated in response to a (17,1) command requesting an Are-You-Alive Connection Test

This packet does not have any parameters.

5.97 TC(17,3) ConnectCmd

Command to perform and On-Board Connection Test.

Table 5.81: ConnectCmd

Name	Byte	Bit	Size	Description
AppId	0	0	8	Identifier of the application with which the connection test must be done
				Total bits: 8 Total bytes: 1.0 Total words: 0.5

5.98 TM(17,4) ConnectRep

Report generated in response to a (17,3) command requesting a On-Board Connection Test

Table 5.82: ConnectRep

Name	Byte	Bit	Size	Description
AppId	0	0	8	Identifier of application with which the connection test was done
				Total bits: 8 Total bytes: 1.0 Total words: 0.5

5.99 TC(255,1) Sample1

Sample command used for testing purposes. The outcome of all its actions and checks can be set by setting user-commandable flags. Its actions increment the value of user-observable counters.

This packet does not have any parameters.

5.100 TM(1,1) SuccAccRep

Report generated to mark the successful acceptance of an incoming command

Table 5.83: SuccAccRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of telecommand being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of telecommand being acknowledged
				Total bits: 32 Total bytes: 4.0 Total words: 2.0

5.101 TM(1,2) FailedAccRep

Report generated to mark the acceptance failure of an incoming command

Table 5.84: FailedAccRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of command being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of command being acknowledged
TcFailCode	4	0	8	Failure Identification Code
TcFailData	5	0	32	Failure data (interpretation depends on the value of the failure code, see description of FailCode)
TcType	9	0	8	Type of Acknowledged TC
TcSubType	10	0	8	Subtype of Acknowledged TC
TcDisc	11	0	16	Discriminant of Acknowledged TC
				Total bits: 104 Total bytes: 13.0 Total words: 6.5

5.102 TM(1,3) SuccStartRep

Report generated to mark the successful start of execution of an incoming command

Table 5.85: SuccStartRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of command being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of command being acknowledged
				Total bits: 32 Total bytes: 4.0 Total words: 2.0

5.103 TM(1,4) FailedStartRep

Report generated to mark the start of execution failure of an incoming command

Table 5.86: FailedStartRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet Version Number
TcPcktId	0	3	13	Packet Identifier of Acknowledged TC
TcPcktSeqCtrl	2	0	16	Packet Seq. Control of Acknowledged TC
TcFailCode	4	0	8	Failure Identification Code
TcFailData	5	0	32	Failure data (interpretation depends on the value of the failure code, see description of FailCode)
TcType	9	0	8	Type of Acknowledged TC
TcSubType	10	0	8	Subtype of Acknowledged TC
TcDisc	11	0	16	Discriminant of Acknowledged TC
				Total bits: 104 Total bytes: 13.0 Total words: 6.5

5.104 TM(1,5) SuccPrgrRep

Report generated to mark the successful completion of an execution step of an incoming command

Table 5.87: SuccPrgrRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of command being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of command being acknowledged
TcPrgStep	4	0	16	Identifier of the progress step which triggered the acknowledge report
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.105 TM(1,6) FailedPrgrRep

Report generated to mark the failure of an execution step of an incoming command

Table 5.88: FailedPrgrRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet Version Number
TcPcktId	0	3	13	Packet Identifier of Acknowledged TC
TcPcktSeqCtrl	2	0	16	Packet Seq. Control of Acknowledged TC
TcFailCode	4	0	8	Failure Identification Code
TcFailData	5	0	32	Failure data (interpretation depends on the value of the failure code, see description of FailCode)
TcType	9	0	8	Type of Acknowledged TC
TcSubType	10	0	8	Subtype of Acknowledged TC
TcDisc	11	0	16	Discriminant of Acknowledged TC
TcPrgStep	13	0	16	Progress step at which the failure was triggered
				Total bits: 120 Total bytes: 15.0 Total words: 7.5

5.106 TM(1,7) SuccTermRep

Report generated to mark the successful completion of execution of an incoming command

Table 5.89: SuccTermRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet version number of command being acknowledged
TcPcktId	0	3	13	Packet identifier of command being acknowledged
TcPcktSeqCtrl	2	0	16	Packet sequence control of command being acknowledged
				Total bits: 32 Total bytes: 4.0 Total words: 2.0

5.107 TM(1,8) FailedTermRep

Report generated to mark the failure to complete execution of an incoming command

Table 5.90: FailedTermRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet Version Number
TcPcktId	0	3	13	Packet Identifier of Acknowledged TC
TcPcktSeqCtrl	2	0	16	Packet Seq. Control of Acknowledged TC
TcFailCode	4	0	8	Failure Identification Code
TcFailData	5	0	32	Failure data (interpretation depends on the value of the failure code, see description of FailCode)
TcType	9	0	8	Type of Acknowledged TC
TcSubType	10	0	8	Subtype of Acknowledged TC
TcDisc	11	0	16	Discriminant of Acknowledged TC
				Total bits: 104 Total bytes: 13.0 Total words: 6.5

5.108 TM(1,10) FailedRoutingRep

Report generated to mark the failure to route an incoming command to its final destination

Table 5.91: FailedRoutingRep

Name	Byte	Bit	Size	Description
PcktVersNumber	0	0	3	Packet Version Number
TcPcktId	0	3	13	Packet Identifier of Acknowledged TC
TcPcktSeqCtrl	2	0	16	Packet Seq. Control of Acknowledged TC
InvDest	4	0	8	invalid Destination for Rerouting Failure
TcType	5	0	8	Type of Acknowledged TC
TcSubType	6	0	8	Subtype of Acknowledged TC
TcDisc	7	0	16	Discriminant of Acknowledged TC
				Total bits: 72 Total bytes: 9.0 Total words: 4.5

5.109 TC(3,1) CreHkCmd

Create a housekeeping report structure

Table 5.92: CreHkCmd

Name	Byte	Bit	Size	Description
SID	0	0	16	The structure identifier (SID) of the packet to be created
CollectionInterval	2	0	16	Collection Interval
N1	4	0	8	The number of parameters in the housekeeping report to be created
- N1ParamId[1]	5	0	16	The identifiers of the simply commutated parameters in the report to be created
- ...				
- N1ParamId[N1]	-	-	16	The identifiers of the simply commutated parameters in the report to be created
NFA	-	-	8	The number of super-commutated groups of parameters
- SCSampleRepNum[1]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[1]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[1][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[1][N2]	-	-	16	Parameter ID
- ...				
- SCSampleRepNum[NFA]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[NFA]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[NFA][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[NFA][N2]	-	-	16	Parameter ID

5.110 TC(3,2) CreDiagCmd

Create a diagnostic report structure

Table 5.93: CreDiagCmd

Name	Byte	Bit	Size	Description
SID	0	0	16	The structure identifier (SID) of the packet to be created
CollectionInterval	2	0	16	Collection Interval
N1	4	0	8	The number of parameters in the diagnostic report to be created
- N1ParamId[1]	5	0	16	The identifiers of the simply commutated parameters in the report to be created
- ...				
- N1ParamId[N1]	-	-	16	The identifiers of the simply commutated parameters in the report to be created
NFA	-	-	8	The number of super-commutated groups of parameters
- SCSampleRepNum[1]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[1]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[1][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[1][N2]	-	-	16	Parameter ID
- ...				
- SCSampleRepNum[NFA]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[NFA]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[NFA][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[NFA][N2]	-	-	16	Parameter ID

5.111 TC(3,3) DelHkCmd

Delete one or more housekeeping report definitions

Table 5.94: DelHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	The number of report definitions to be deleted
- SID[1]	1	0	16	The structure identifiers (SIDs) of the report definitions to be deleted
- ...				
- SID[N]	-	-	16	The structure identifiers (SIDs) of the report definitions to be deleted

5.112 TC(3,4) DelDiagCmd

Delete one or more diagnostic report definitions

Table 5.95: DelDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	The number of report definitions to be deleted
- SID[1]	1	0	16	The structure identifier (SID) of a report to be deleted
- ...				
- SID[N]	-	-	16	The structure identifier (SID) of a report to be deleted

5.113 TC(3,5) EnbHkCmd

Enable the periodic generation of one or more housekeeping report structures

Table 5.96: EnbHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be enabled
- SID[1]	1	0	16	SID to be enabled
- ...				
- SID[N]	-	-	16	SID to be enabled

5.114 TC(3,6) DisHkCmd

Disable the periodic generation of one or more housekeeping report structures

Table 5.97: DisHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be disabled
- SID[1]	1	0	16	SID to be disabled
- ...				
- SID[N]	-	-	16	SID to be disabled

5.115 TC(3,7) EnbDiagCmd

Enable the periodic generation of one or more diagnostic report structures

Table 5.98: EnbDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be enabled
- SID[1]	1	0	16	SID to be enabled
- ...				
- SID[N]	-	-	16	SID to be enabled

5.116 TC(3,8) DisDiagCmd

Disable the periodic generation of one or more diagnostic report structures

Table 5.99: DisDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be disabled
- SID[1]	1	0	16	SID to be disabled
- ...				
- SID[N]	-	-	16	SID to be disabled

5.117 TC(3,9) RepStructHkCmd

This command carries a list of SIDs. For each SID, it triggers the generation of a (3,10) report with the definition of the housekeeping report structure for that SID.

Table 5.100: RepStructHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be reported
- SID[1]	1	0	16	SID to be reported
- ...				
- SID[N]	-	-	16	SID to be reported

5.118 TM(3,10) RepStructHkRep

Report carrying the definition of a housekeeping report structure generated in response to a (3,9) command.

Table 5.101: RepStructHkRep

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
PerGenActionStatus	2	0	8	Flag indicating whether periodic generation of the packet is enabled or disabled
CollectionInterval	3	0	16	Collection Interval
N1	5	0	8	The number of simply commutated parameters
- N1ParamId[1]	6	0	16	Identifier of a simply commutated parameter
- ...				
- N1ParamId[N1]	-	-	16	Identifier of a simply commutated parameter
NFA	-	-	8	The number of super-commutated groups of parameters
- SCSampleRepNum[1]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[1]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[1][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[1][N2]	-	-	16	Parameter ID
- ...				
- SCSampleRepNum[NFA]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[NFA]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[NFA][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[NFA][N2]	-	-	16	Parameter ID

5.119 TC(3,11) RepStructDiagCmd

This command carries a list of SIDs. For each SID, it triggers the generation of a (3,12) report with the definition of the diagnostic report structure for that SID.

Table 5.102: RepStructDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be reported
- SID[1]	1	0	16	SID to be reported
- ...				
- SID[N]	-	-	16	SID to be reported

5.120 TM(3,12) RepStructDiagRep

Report carrying the definition of a diagnostic report structure generated in response to a (3,11) command.

Table 5.103: RepStructDiagRep

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
PerGenActionStatus	2	0	8	Flag indicating whether periodic generation of the packet is enabled or disabled
CollectionInterval	3	0	16	Collection Interval
N1	5	0	8	The number of simply commutated parameters
- N1ParamId[1]	6	0	16	Identifier of a simply commutated parameter
- ...				
- N1ParamId[N1]	-	-	16	Identifier of a simply commutated parameter
NFA	-	-	8	The number of super-commutated groups of parameters
- SCSampleRepNum[1]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[1]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[1][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[1][N2]	-	-	16	Parameter ID
- ...				
- SCSampleRepNum[NFA]	-	-	8	Super Commutated Sample Repetition Number (repeated NFA times)
- N2[NFA]	-	-	8	The number of parameters in the super-commutated group
- - N2ParamId[NFA][1]	-	-	16	Parameter ID
- - ...				
- - N2ParamId[NFA][N2]	-	-	16	Parameter ID

5.121 TM(3,25) Rep (SID_HK_CNT)

Housekeeping packet holding the cycle counters of the HK packets

Table 5.104: Rep (SID_HK_CNT)

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
cycleCnt	2	0	4*16	Cycle Counter for Reports in RDL
				Total bits: 80 Total bytes: 10.0 Total words: 5.0

5.122 TM(3,25) Rep (SID_N_OF_EVT)

Housekeeping packet holding number of generated events of each severity level

Table 5.105: Rep (SID_N_OF_EVT)

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
nOfDetectedEvs	2	0	4*16	Number of detected occurrences of events (one element for each severity level)
nOfDisabledEid	10	0	4*16	Number of disabled event identifiers (one element for each severity level)
				Total bits: 144 Total bytes: 18.0 Total words: 9.0

5.123 TM(3,26) DiagRep

Periodic Diagnostic Report (3,26)

Table 5.106: DiagRep

Name	Byte	Bit	Size	Description
SID	0	0	16	Structure Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.124 TC(3,27) OneShotHkCmd

Command (3,27) to generate a one-shot housekeeping report

Table 5.107: OneShotHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be generated in one-shot mode
- SID[1]	1	0	16	SID to be generated in one-shot mode
- ...				
- SID[N]	-	-	16	SID to be generated in one-shot mode

5.125 TC(3,28) OneShotDiagCmd

Command (3,28) to generate a one-shot diagnostic report

Table 5.108: OneShotDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs to be generated in one-shot mode
- SID[1]	1	0	16	SID to be generated in one-shot mode
- ...				
- SID[N]	-	-	16	SID to be generated in one-shot mode

5.126 TC(3,31) ModPerHkCmd

Command (3,31) to modify the collection period of a housekeeping report

Table 5.109: ModPerHkCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs whose collection interval is to be modified
- SID[1]	1	0	16	SID whose collection interval is to be modified
- CollectionInterval[1]	3	0	16	New collection interval
- ...				
- SID[N]	-	-	16	SID whose collection interval is to be modified
- CollectionInterval[N]	-	-	16	New collection interval

5.127 TC(3,32) ModPerDiagCmd

Command (3,31) to modify the collection period of a diagnostic report

Table 5.110: ModPerDiagCmd

Name	Byte	Bit	Size	Description
N	0	0	8	Number of SIDs whose collection interval is to be modified
- SID[1]	1	0	16	SID whose collection interval is to be modified
- CollectionInterval[1]	3	0	16	New collection interval
- ...				
- SID[N]	-	-	16	SID whose collection interval is to be modified
- CollectionInterval[N]	-	-	16	New collection interval

5.128 TM(5,1) Rep1 (EVT_DOWN_ABORT)

Triggered by the aborting of a down-transfer

Table 5.111: Rep1 (EVT_DOWN_ABORT)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
LptSmId	2	0	8	Identifier of LPT State Machine where the down-transfer abort occurred
				Total bits: 24 Total bytes: 3.0 Total words: 1.5

5.129 TM(5,1) Rep1 (EVT_DUMMY_1)

Dummy event with one dummy parameter used for testing

Table 5.112: Rep1 (EVT_DUMMY_1)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
Par	2	0	8	Dummy parameter for dummy event used for testing
				Total bits: 24 Total bytes: 3.0 Total words: 1.5

5.130 TM(5,1) Rep1 (EVT_UP_ABORT)

Triggered by the aborting of a up-transfer

Table 5.113: Rep1 (EVT_UP_ABORT)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
LptSmId	2	0	8	Identifier of LPT State Machine where the up-transfer abort occurred
				Total bits: 24 Total bytes: 3.0 Total words: 1.5

5.131 TM(5,2) Rep2 (EVT_CLST_FULL)

Generated when the Monitoring Function Procedure tries to add an entry to the Check Transition List but the list is full

Table 5.114: Rep2 (EVT_CLST_FULL)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.132 TM(5,2) Rep2 (EVT_DUMMY_2)

Dummy level 2 event (no parameters)

Table 5.115: Rep2 (EVT_DUMMY_2)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.133 TM(5,3) Rep3 (EVT_DUMMY_3)

Dummy level 3 event (one dummy parameter)

Table 5.116: Rep3 (EVT_DUMMY_3)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
Par	2	0	8	Dummy parameter for dummy level 3 event
				Total bits: 24 Total bytes: 3.0 Total words: 1.5

5.134 TM(5,3) Rep3 (EVT_FMON_FAIL)

Generated when a functional monitor has declared a failure

Table 5.117: Rep3 (EVT_FMON_FAIL)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.135 TM(5,3) Rep3 (EVT_MON_DEL_I)

Generated when a Delta Check Monitoring Procedure has detected an invalid parameter value of integer type

Table 5.118: Rep3 (EVT_MON_DEL_I)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
ParMonId	2	0	16	The identifier of the parameter monitor which reported the violation
MonParId	4	0	16	The identifier of the monitored parameter which went out of limits
ParMonCheckStatus	6	0	8	Last status of parameter monitor which reported limit violation
ParValueInt	7	0	32	Last value of monitored parameter which went out-of-limits
				Total bits: 88 Total bytes: 11.0 Total words: 5.5

5.136 TM(5,3) Rep3 (EVT_MON_DEL_R)

Generated when a Delta Check Monitoring Procedure has detected an invalid parameter value of real type

Table 5.119: Rep3 (EVT_MON_DEL_R)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
ParMonId	2	0	16	The identifier of the parameter monitor which reported the violation
MonParId	4	0	16	The identifier of the monitored parameter which went out of limits
ParMonCheckStatus	6	0	8	Last status of parameter monitor which reported limit violation
ParValueReal	7	0	32	Last value of monitored parameter which went out-of-limits
				Total bits: 88 Total bytes: 11.0 Total words: 5.5

5.137 TM(5,3) Rep3 (EVT_MON_LIM_I)

Generated when a Limit Check Monitoring Procedure has detected an invalid parameter value of integer type

Table 5.120: Rep3 (EVT_MON_LIM_I)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
ParMonId	2	0	16	The identifier of the parameter monitor which reported the violation
MonParId	4	0	16	The identifier of the monitored parameter which went out of limits
ParMonCheckStatus	6	0	8	Last status of parameter monitor which reported limit violation
ParValueInt	7	0	32	Last value of monitored parameter which went out-of-limits
				Total bits: 88 Total bytes: 11.0 Total words: 5.5

5.138 TM(5,3) Rep3 (EVT_MON_LIM_R)

Generated when a Limit Check Monitoring Procedure has detected an invalid parameter value of real type

Table 5.121: Rep3 (EVT_MON_LIM_R)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
ParMonId	2	0	16	The identifier of the parameter monitor which reported the violation
MonParId	4	0	16	The identifier of the monitored parameter which went out of limits
ParMonCheckStatus	6	0	8	Last status of parameter monitor which reported limit violation
ParValueReal	7	0	32	Last value of monitored parameter which went out-of-limits
				Total bits: 88 Total bytes: 11.0 Total words: 5.5

5.139 TM(5,4) Rep4 (EVT_DUMMY_4)

Dummy level 4 event (no parameters)

Table 5.122: Rep4 (EVT_DUMMY_4)

Name	Byte	Bit	Size	Description
EventId	0	0	16	Event Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.140 TC(5,5) EnbCmd

Command to enable generation of a list of event identifiers

Table 5.123: EnbCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of event identifiers to be enabled
- EventId[1]	2	0	16	Event identifier to be enabled
- ...				
- EventId[N]	-	-	16	Event identifier to be enabled

5.141 TC(5,6) DisCmd

Command to disable generation of a list of event identifiers

Table 5.124: DisCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of event identifiers to be disabled
- EventId[1]	2	0	16	Event identifier to be disabled
- ...				
- EventId[N]	-	-	16	Event identifier to be disabled

5.142 TC(5,7) RepDisCmd

This command triggers the generation of a (5,8) report holding the list of disabled event identifiers

This packet does not have any parameters.

5.143 TM(5,8) DisRep

Report generated in response to a (5,7) command carrying the list of disabled Event Identifiers

Table 5.125: DisRep

Name	Byte	Bit	Size	Description
N	0	0	16	The number of disabled event identifiers
- EventId[1]	2	0	16	Event Identifier
- EventId[1]	4	0	16	Event Identifier
- ...				
- EventId[N]	-	-	16	Event Identifier
- EventId[N]	-	-	16	Event Identifier

5.144 TC(11,1) EnbTbsCmd

Command to enable the time-based schedule execution function

This packet does not have any parameters.

5.145 TC(11,2) DisTbsCmd

Command to disable the time-based schedule execution function

This packet does not have any parameters.

5.146 TC(11,3) ResTbsCmd

Command to reset the time-based schedule

This packet does not have any parameters.

5.147 TC(11,4) InsTbaCmd

Command to insert one or more time-based activities (TBAs) into the time-based schedule (TBS)

Table 5.126: InsTbaCmd

Name	Byte	Bit	Size	Description
SubSchedId	0	0	8	Sub-schedule to which the activities are assigned
N	1	0	16	Number of activities to be inserted in the time-based schedule
- GroupId[1]	3	0	8	Schedule group to which this activity is assigned
- RelTime[1]	4	0	48	Release time of activity
- Request[1]	10	0	undefined	Command which implements the activity
- ...				
- GroupId[N]	-	-	8	Schedule group to which this activity is assigned
- RelTime[N]	-	-	48	Release time of activity
- Request[N]	-	-	undefined	Command which implements the activity

5.148 TC(11,5) DelTbaCmd

Command to delete one or more time-based activities (TBAs) from the time-based schedule (TBS)

Table 5.127: DelTbaCmd

Name	Byte	Bit	Size	Description
N	0	0	16	Number of activities to be inserted in the time-based schedule
- AppId[1]	2	0	8	Source of command embedded in activity to be deleted
- APID[1]	3	0	11	APID of command embedded in activity to be deleted
- SrcSeqCnt[1]	4	3	14	Source sequence count of command embedded in activity to be deleted
- ...				
- AppId[N]	-	-	8	Source of command embedded in activity to be deleted
- APID[N]	-	-	11	APID of command embedded in activity to be deleted
- SrcSeqCnt[N]	-	-	14	Source sequence count of command embedded in activity to be deleted

5.149 TC(11,20) EnbSubSchedCmd

Command to enable one or more time-based sub-schedules

Table 5.128: EnbSubSchedCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of sub-schedule identifiers to be enabled
- SubSchedId[1]	2	0	8	The identifier of a sub-schedule to be enabled
- ...				
- SubSchedId[N]	-	-	8	The identifier of a sub-schedule to be enabled

5.150 TC(11,21) DisSubSchedCmd

Command to disable one or more time-based sub-schedules

Table 5.129: DisSubSchedCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of sub-schedule identifiers to be disabled
- SubSchedId[1]	2	0	8	The identifier of a sub-schedule to be disabled
- ...				
- SubSchedId[N]	-	-	8	The identifier of a sub-schedule to be disabled

5.151 TC(11,22) CreGrpCmd

Command to create one or more scheduling groups

Table 5.130: CreGrpCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of groups to be created
- GroupId[1]	2	0	8	The identifier of a group to be created
- isGroupEnabled[1]	3	0	8	The initial enable status of the group
- ...				
- GroupId[N]	-	-	8	The identifier of a group to be created
- isGroupEnabled[N]	-	-	8	The initial enable status of the group

5.152 TC(11,23) DelGrpCmd

Command to delete one or more scheduling groups

Table 5.131: DelGrpCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of groups to be deleted
- GroupId[1]	2	0	8	The identifier of a group to be deleted
- ...				
- GroupId[N]	-	-	8	The identifier of a group to be deleted

5.153 TC(11,24) EnbGrpCmd

Command to enable one or more scheduling groups

Table 5.132: EnbGrpCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of groups to be enabled (if this is zero, then all groups are enabled)
- GroupId[1]	2	0	8	The identifier of a group to be enabled
- ...				
- GroupId[N]	-	-	8	The identifier of a group to be enabled

5.154 TC(11,25) DisGrpCmd

Command to disable one or more scheduling groups

Table 5.133: DisGrpCmd

Name	Byte	Bit	Size	Description
N	0	0	16	The number of groups to be enabled (if this is zero, then all groups currently in use are disabled)
- GroupId[1]	2	0	8	The identifier of a group to be enabled
- ...				
- GroupId[N]	-	-	8	The identifier of a group to be enabled

5.155 TC(11,26) RepGrpCmd

Command to trigger the generation of a (11,27) report carrying the status of the scheduling groups

This packet does not have any parameters.

5.156 TM(11,27) GrpRep

Report generated in response to a (11,26) command to report the status of the scheduling groups

Table 5.134: GrpRep

Name	Byte	Bit	Size	Description
N	0	0	16	Number of groups being reported
- GroupId[1]	2	0	8	The identifier of a group being reported
- isGroupEnabled[1]	3	0	8	The enable status of a group being reported
- ...				
- GroupId[N]	-	-	8	The identifier of a group being reported
- isGroupEnabled[N]	-	-	8	The enable status of a group being reported

5.157 TC(12,1) EnbParMonDefCmd

Command to enable one or more monitoring definitions

Table 5.135: EnbParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of monitoring definitions to be enabled.
- ParMonId[1]	1	0	16	Identifier of Parameter Monitor
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of Parameter Monitor

5.158 TC(12,2) DisParMonDefCmd

Command to disable one or more monitoring definitions

Table 5.136: DisParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of monitoring definitions to be disabled,
- ParMonId[1]	1	0	16	Identifier of Parameter Monitor
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of Parameter Monitor

5.159 TC(12,3) ChgTransDelCmd

Command to change the maximum delay after which the content of the check transition list (CTL) is reported through a (12,12) report

Table 5.137: ChgTransDelCmd

Name	Byte	Bit	Size	Description
maxRepDelay	0	0	16	Maximum reporting delay
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.160 TC(12,4) DelAllParMonCmd

Command to delete all parameter monitoring definitions

This packet does not have any parameters.

5.161 TC(12,5) AddParMonDefCmd

Command to add one or more parameter definitions

Table 5.138: AddParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of parameter definitions
- ParMonId[1]	1	0	16	Identifier of parameter monitor to be added by telecommand
- MonParId[1]	3	0	16	Identifier of Monitored Parameter
- ValCheckParId[1]	5	0	16	Identifier of Validity Parameter
- ExpValCheckMask[1]	7	0	16	Expected Value Check Mask
- ValCheckExpVal[1]	9	0	undefined	Expected Value for Validity Check
- CheckTypeData[1]	-	-	undefined	For expected value check, this parameter consists of: mask, expected value and event identifier. For limit checks, it consists of: low limit, event identifier for low limit, high limit, event identifier for high limit. For delta checks, it consists of: low delta threshold, event identifier for low threshold, high delta threshold, event identifiers for high thresholds. In all cases an event identifier of zero indicates that no event is associated to the violation.
- RepNmb[1]	-	-	8	Repetition Number
- CheckType[1]	-	-	8	Type of Monitor Procedure
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of parameter monitor to be added by telecommand
- MonParId[NParMon]	-	-	16	Identifier of Monitored Parameter
- ValCheckParId[NParMon]	-	-	16	Identifier of Validity Parameter
- ExpValCheckMask[NParMon]	-	-	16	Expected Value Check Mask
- ValCheckExpVal[NParMon]	-	-	undefined	Expected Value for Validity Check
- CheckTypeData[NParMon]	-	-	undefined	For expected value check, this parameter consists of: mask, expected value and event identifier. For limit checks, it consists of: low limit, event identifier for low limit, high limit, event identifier for high limit. For delta checks, it consists of: low delta threshold, event identifier for low threshold, high delta threshold, event identifiers for high thresholds. In all cases an event identifier of zero indicates that no event is associated to the violation.

Name	Byte	Bit	Size	Description
- RepNmb[NParMon]	-	-	8	Repetition Number
- CheckType[NParMon]	-	-	8	Type of Monitor Procedure
CheckTypeData	-	-	undefined	For expected value check, this parameter consists of: mask, expected value and event identifier. For limit checks, it consists of: low limit, event identifier for low limit, high limit, event identifier for high limit. For delta checks, it consists of: low delta threshold, event identifier for low threshold, high delta threshold, event identifiers for high thresholds. In all cases, an event identifier of zero indicates that no event is associated to the violation.

5.162 TC(12,6) DelParMonDefCmd

Command to delete one or more parameter monitoring definitions

Table 5.139: DelParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of parameter monitoring definitions to be deleted
- ParMonId[1]	1	0	16	The identifier of the parameter monitoring definition to be deleted
- ...				
- ParMonId[NParMon]	-	-	16	The identifier of the parameter monitoring definition to be deleted

5.163 TC(12,7) ModParMonDefCmd

Command to modify one or more parameter definitions

Table 5.140: ModParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of parameter monitoring definitions to be modified
MonParId	1	0	16	The identifier of the parameter monitor to be modified
MonParId	3	0	16	The identifier of the data pool item to be monitored by the modified parameter monitor
RepNmb	5	0	8	The repetition number of the modified parameter monitor
CheckType	6	0	8	The monitoring check type of the modified parameter monitor
CheckTypeData	7	0	undefined	For expected value check, this parameter consists of: mask, expected value and event identifier. For limit checks, it consists of: low limit, event identifier for low limit, high limit, event identifier for high limit. For delta checks, it consists of: low delta threshold, event identifier for low threshold, high delta threshold, event identifiers for high thresholds. In all cases, an event identifier of zero indicates that no event is associated to the violation.

5.164 TC(12,8) RepParMonDefCmd

This command triggers the generation of a (12,9) report carrying one or more parameter monitor definitions

Table 5.141: RepParMonDefCmd

Name	Byte	Bit	Size	Description
NParMon	0	0	8	The number of parameter monitoring definitions to be reported or zero if all parameter monitoring definitions must be reported
- ParMonId[1]	1	0	16	Identifier of a parameter monitor to be reported
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of a parameter monitor to be reported

5.165 TM(12,9) RepParMonDefRep

Report generated in response to a (12,8) command to report one or more monitoring definitions.

This packet does not have any parameters.

5.166 TC(12,10) RepOutOfLimitsCmd

This command triggers the generation of a (12,11) report holding the parameter monitors which are out of limits

This packet does not have any parameters.

5.167 TM(12,11) RepOutOfLimitsRep

Report generated in response to a (12,10) command carrying the parameter monitors which are out of limits

Table 5.142: RepOutOfLimitsRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of out of limit parameter monitors which are reported
- MonParId[1]	1	0	16	Identifier of data pool item monitored by parameter monitor
- ParMonId[1]	3	0	16	Identifier of Parameter Monitor
- CheckType[1]	5	0	8	Type of Monitor Procedure
- ParValue[1]	6	0	undefined	Parameter Value at Monitoring Violation
- LimitCrossed[1]	-	-	undefined	Limit Crossed at Monitoring Violation
- ParMonCheckStatus[1]	-	-	8	Parameter Monitor Checking Status
- ParMonPrevCheckStatus[1]	-	-	8	Checking Status Before Monitoring Violation
- TransTime[1]	-	-	48	Time of Monitoring Violation Transition
- ...				
- MonParId[NParMon]	-	-	16	Identifier of data pool item monitored by parameter monitor
- ParMonId[NParMon]	-	-	16	Identifier of Parameter Monitor
- CheckType[NParMon]	-	-	8	Type of Monitor Procedure
- ParValue[NParMon]	-	-	undefined	Parameter Value at Monitoring Violation
- LimitCrossed[NParMon]	-	-	undefined	Limit Crossed at Monitoring Violation
- ParMonCheckStatus[NParMon]	-	-	8	Parameter Monitor Checking Status
- ParMonPrevCheckStatus[NParMon]	-	-	8	Checking Status Before Monitoring Violation
- TransTime[NParMon]	-	-	48	Time of Monitoring Violation Transition

5.168 TM(12,12) CheckTransRep

Report carrying the content of the Check Transition List (CTL).

Table 5.143: CheckTransRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of check transitions in the report
- ParMonId[1]	1	0	16	Identifier of Parameter Monitor
- MonParId[1]	3	0	16	Identifier of Monitored Parameter
- CheckType[1]	5	0	8	Type of Monitor Procedure
- ExpValCheckMask[1]	6	0	16	Expected Value Check Mask
- ParValue[1]	8	0	undefined	Parameter Value at Monitoring Violation
- LimitCrossed[1]	-	-	undefined	Limit Crossed at Monitoring Violation
- ParMonPrevCheckStatus[1]	-	-	8	Checking Status Before Monitoring Violation
- ParMonCheckStatus[1]	-	-	8	Parameter Monitor Checking Status
- TransTime[1]	-	-	48	Time of Monitoring Violation Transition
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of Parameter Monitor
- MonParId[NParMon]	-	-	16	Identifier of Monitored Parameter
- CheckType[NParMon]	-	-	8	Type of Monitor Procedure
- ExpValCheckMask[NParMon]	-	-	16	Expected Value Check Mask
- ParValue[NParMon]	-	-	undefined	Parameter Value at Monitoring Violation
- LimitCrossed[NParMon]	-	-	undefined	Limit Crossed at Monitoring Violation
- ParMonPrevCheckStatus[NParMon]	-	-	8	Checking Status Before Monitoring Violation
- ParMonCheckStatus[NParMon]	-	-	8	Parameter Monitor Checking Status
- TransTime[NParMon]	-	-	48	Time of Monitoring Violation Transition

5.169 TC(12,13) RepParMonStatCmd

This command triggers the generation of a (12,14) report carrying the status of all parameter monitors

This packet does not have any parameters.

5.170 TM(12,14) RepParMonStatRep

Report generated in response to a (12,13) report carrying the status of all currently defined parameter monitors

Table 5.144: RepParMonStatRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	Number of parameter monitors whose status is reported by the telecommand
- ParMonId[1]	1	0	16	Identifier of a parameter monitor
- ParMonCheckStatus[1]	3	0	8	Current checking status of the parameter monitor
- ...				
- ParMonId[NParMon]	-	-	16	Identifier of a parameter monitor
- ParMonCheckStatus[NParMon]	-	-	8	Current checking status of the parameter monitor

5.171 TC(12,15) EnbParMonFuncCmd

Command to enable the monitoring function

This packet does not have any parameters.

5.172 TC(12,16) DisParMonFuncCmd

Command to disable the parameter monitoring function

This packet does not have any parameters.

5.173 TC(12,17) EnbFuncMonCmd

Command to enable the functional monitoring function

This packet does not have any parameters.

5.174 TC(12,18) DisFuncMonCmd

Command to disable the functional monitoring function

This packet does not have any parameters.

5.175 TC(12,19) EnbFuncMonDefCmd

Command to enable one or more functional monitoring definitions

Table 5.145: EnbFuncMonDefCmd

Name	Byte	Bit	Size	Description
NFuncMon	0	0	8	The number of functional monitoring definitions to be enabled
- FuncMonId[1]	1	0	8	Identifier of a functional monitor to be enabled
- ...				
- FuncMonId[NFuncMon]	-	-	8	Identifier of a functional monitor to be enabled

5.176 TC(12,20) DisFuncMonDefCmd

Command to disable one ore more functional monitoring definitions

Table 5.146: DisFuncMonDefCmd

Name	Byte	Bit	Size	Description
NFuncMon	0	0	8	The number of functional monitors to be disabled
- FuncMonId[1]	1	0	8	Identifer of functional monitor to be disabled
- ...				
- FuncMonId[NFuncMon]	-	-	8	Identifer of functional monitor to be disabled

5.177 TC(12,21) ProtFuncMonDefCmd

Command to protect one or more functional monitoring definitions

Table 5.147: ProtFuncMonDefCmd

Name	Byte	Bit	Size	Description
NFuncMon	0	0	8	The number of functional monitors to be protected
- FuncMonId[1]	1	0	8	Identifier of functional monitor to be protected
- ...				
- FuncMonId[NFuncMon]	-	-	8	Identifier of functional monitor to be protected

5.178 TC(12,22) UnprotFuncMonDefCmd

Command to unprotect one or more functional monitoring definitions

Table 5.148: UnprotFuncMonDefCmd

Name	Byte	Bit	Size	Description
FuncMonId	0	0	8	The number of functional monitor to be unprotected
- FuncMonId[1]	1	0	8	Identifier of a functional monitor to be unprotected
- ...				
- FuncMonId[FuncMonId]	-	-	8	Identifier of a functional monitor to be unprotected

5.179 TC(12,23) AddFuncMonDefCmd

Command to add one or more functional monitoring definitions

Table 5.149: AddFuncMonDefCmd

Name	Byte	Bit	Size	Description
FuncMonId	0	0	8	Number of functional monitor definitions to be added
- FuncMonId[1]	1	0	8	Identifier of functional monitor to be added
- ValCheckParId[1]	2	0	16	Identifier of Validity Parameter
- ValCheckParMask[1]	4	0	undefined	Mask for Validity Check
- ValCheckExpVal[1]	-	-	undefined	Expected Value for Validity Check
- EvtId[1]	-	-	16	Event Identifier
- MinFailNmb[1]	-	-	8	Minimum Failing Number
- NFuncMon[1]	-	-	8	Number of parameter monitors attached to functional monitor
- ParMonId[1]	-	-	16	Identifier of Parameter Monitor
- ...				
- FuncMonId[FuncMonId]	-	-	8	Identifier of functional monitor to be added
- ValCheckParId[FuncMonId]	-	-	16	Identifier of Validity Parameter
- ValCheckParMask[FuncMonId]	-	-	undefined	Mask for Validity Check
- ValCheckExpVal[FuncMonId]	-	-	undefined	Expected Value for Validity Check
- EvtId[FuncMonId]	-	-	16	Event Identifier
- MinFailNmb[FuncMonId]	-	-	8	Minimum Failing Number
- NFuncMon[FuncMonId]	-	-	8	Number of parameter monitors attached to functional monitor
- ParMonId[FuncMonId]	-	-	16	Identifier of Parameter Monitor

5.180 TC(12,24) DelFuncMonDefCmd

Command to delete one or more functional monitoring definitions to the FMDL

Table 5.150: DelFuncMonDefCmd

Name	Byte	Bit	Size	Description
NFuncMon	0	0	8	Number of functional monitors to be deleted
- FuncMonId[1]	1	0	8	Identifier of functional monitor to be deleted
- ...				
- FuncMonId[NFuncMon]	-	-	8	Identifier of functional monitor to be deleted

5.181 TC(12,25) RepFuncMonDefCmd

This command triggers the generation of a (12,26) report carrying the definition of one or more functional monitors

Table 5.151: RepFuncMonDefCmd

Name	Byte	Bit	Size	Description
FuncMonId	0	0	8	Number of functional monitors whose definition is to be reported
- FuncMonId[1]	1	0	8	Identifier of functional monitor whose definition is to be reported
- ...				
- FuncMonId[FuncMonId]	-	-	8	Identifier of functional monitor whose definition is to be reported

5.182 TM(12,26) RepFuncMonDefRep

Report generated in response to a (12,25) command to carry the definition of some or all functional monitoring definitions

Table 5.152: RepFuncMonDefRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	The number of functional monitoring definitions in the report
- FuncMonId[1]	1	0	8	Identifier of Functional Monitor
- ValCheckParId[1]	2	0	16	Identifier of Validity Parameter
- ValCheckParMask[1]	4	0	undefined	Mask for Validity Check
- ValCheckParMask[1]	-	-	undefined	Mask for Validity Check
- ProtStatus[1]	-	-	8	Functional Monitor Protection Status
- FuncMonCheckStatus[1]	-	-	8	Functional Monitor Checking Status
- EvtId[1]	-	-	16	Event Identifier
- MinFailNmb[1]	-	-	8	Minimum Failing Number
- NFuncMon[1]	-	-	8	The number of parameter monitors associated to the functional monitor
- - ParMonId[1][1]	-	-	16	Identifier of Parameter Monitor
- - ...				
- - ParMonId[1][NFuncMon]	-	-	16	Identifier of Parameter Monitor
- ...				
- FuncMonId[NParMon]	-	-	8	Identifier of Functional Monitor
- ValCheckParId[NParMon]	-	-	16	Identifier of Validity Parameter
- ValCheckParMask[NParMon]	-	-	undefined	Mask for Validity Check
- ValCheckParMask[NParMon]	-	-	undefined	Mask for Validity Check
- ProtStatus[NParMon]	-	-	8	Functional Monitor Protection Status
- FuncMonCheckStatus[NParMon]	-	-	8	Functional Monitor Checking Status
- EvtId[NParMon]	-	-	16	Event Identifier
- MinFailNmb[NParMon]	-	-	8	Minimum Failing Number
- NFuncMon[NParMon]	-	-	8	The number of parameter monitors associated to the functional monitor

Name	Byte	Bit	Size	Description
- - ParMonId[NParMon][1]	-	-	16	Identifier of Parameter Monitor
- - ...				
- - ParMonId[NParMon][NFuncMon]	-	-	16	Identifier of Parameter Monitor

5.183 TC(12,27) RepFuncMonStatCmd

This command triggers the generation of a (12,28) report carrying the status of all functional monitors

This packet does not have any parameters.

5.184 TM(12,28) RepFuncMonStatRep

Report generated in response to a (12,27) command carrying the status of all currently defined functional monitors

Table 5.153: RepFuncMonStatRep

Name	Byte	Bit	Size	Description
NParMon	0	0	8	The number of functional monitor statuses in the report
- FuncMonId[1]	1	0	8	Identifier of Functional Monitor
- ProtStatus[1]	2	0	8	Functional Monitor Protection Status
- IsEnabled[1]	3	0	8	Enable Status
- FuncMonCheckStatus[1]	4	0	8	Functional Monitor Checking Status
- ...				
- FuncMonId[NParMon]	-	-	8	Identifier of Functional Monitor
- ProtStatus[NParMon]	-	-	8	Functional Monitor Protection Status
- IsEnabled[NParMon]	-	-	8	Enable Status
- FuncMonCheckStatus[NParMon]	-	-	8	Functional Monitor Checking Status

5.185 TM(13,1) DownFirstRep

Report carrying the first part of a down-transfer

Table 5.154: DownFirstRep

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Down-transfer data (repetition value to be set to fill the packet)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.186 TM(13,2) DownInterRep

Report carrying an intermediate part of a down-transfer

Table 5.155: DownInterRep

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Down-transfer data (repetition value to be set to fill the packet)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.187 TM(13,3) DownLastRep

Report carrying the last part of a down-transfer

Table 5.156: DownLastRep

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Down-transfer data (repetition value is set dynamically by the host application based on the amount of data to be down-transferred)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.188 TC(13,9) UpFirstCmd

Command to carry the first part of an up-transfer

Table 5.157: UpFirstCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Up-transfer data (repetition value to be set to fill the packet)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.189 TC(13,10) UpInterCmd

Command to carry an intermediate part of an up-transfer

Table 5.158: UpInterCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Up-transfer data (repetition value to be set to fill the packet)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.190 TC(13,11) UpLastCmd

Command to carry the last part of an up-transfer

Table 5.159: UpLastCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
PartSeqNmb	2	0	16	Part Sequence Number
Part	4	0	16	Up-transfer data (repetition value is set dynamically)
				Total bits: 48 Total bytes: 6.0 Total words: 3.0

5.191 TM(13,16) UpAbortRep

Report to notify the abortion of an up-transfer

Table 5.160: UpAbortRep

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
FailReason	2	0	16	Transfer Failure Reason
				Total bits: 32 Total bytes: 4.0 Total words: 2.0

5.192 TC(13,129) StartDownCmd

Command to start a down-transfer

Table 5.161: StartDownCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.193 TC(13,130) AbortDownCmd

Command to abort a down-transfer

Table 5.162: AbortDownCmd

Name	Byte	Bit	Size	Description
Transid	0	0	16	Large Message Trans. Identifier
				Total bits: 16 Total bytes: 2.0 Total words: 1.0

5.194 TC(17,1) AreYouAliveCmd

Command to perform and Are-You-Alive Connection Test

This packet does not have any parameters.

5.195 TM(17,2) AreYouAliveRep

Report generated in response to a (17,1) command requesting an Are-You-Alive Connection Test

This packet does not have any parameters.

5.196 TC(17,3) ConnectCmd

Command to perform and On-Board Connection Test.

Table 5.163: ConnectCmd

Name	Byte	Bit	Size	Description
AppId	0	0	8	Identifier of the application with which the connection test must be done
				Total bits: 8 Total bytes: 1.0 Total words: 0.5

5.197 TM(17,4) ConnectRep

Report generated in response to a (17,3) command requesting a On-Board Connection Test

Table 5.164: ConnectRep

Name	Byte	Bit	Size	Description
AppId	0	0	8	Identifier of application with which the connection test was done
				Total bits: 8 Total bytes: 1.0 Total words: 0.5

5.198 TC(255,1) Sample1

Sample command used for testing purposes. The outcome of all its actions and checks can be set by setting user-commandable flags. Its actions increment the value of user-observable counters.

This packet does not have any parameters.

6 Data Pool Definition

The data pool is split into two parts of which one holds the variables and the other holds the parameters. The first two tables in this section list, respectively, the data pool variables and the data pool parameters. The last table lists the constants and their values (some of these constants are used to express the multiplicity of variables or parameters in the data pool).

Table 6.1: Datapool Parameters

DPID	Name	Description	Default	Type	Size
0x1	debugVarAddr	Address of Debug Variables	0	CrPsThirtytwoBit_t[HK_N_DEBUG_VAR]	96
0x2	dest	Destination of report definitions in the RDL	0	CrFwDestSrc_t[HK_N_REP_DEF]	32
0x3	isEnabled	Enable status of report definitions in the RDL	0	CrFwBool_t[HK_N_REP_DEF]	32
0x4	nSimple	Number of simply commutated data items in HK report in RDL	0	CrPsNPar_t[HK_N_REP_DEF]	32
0x5	period	Periods of report definitions in the RDL	0	CrPsCycleCnt_t[HK_N_REP_DEF]	64
0x6	sid	SIDs of report definitions in the RDL	0	CrPsSID_t[HK_N_REP_DEF]	64
0x7	lptDest	Destination of transfer from LPT Buffer	0	CrFwDestSrc_t[LPT_N_BUF]	8
0x8	lptTimeOut	Time-out for up-transfer to LPT Buffer	0	CrPsTimeSec_t[LPT_N_BUF]	32
0x9	partSize	Part size for transfers to/from LPT Buffer	0	CrPsSize_t[LPT_N_BUF]	16
0xa	maxRepDelay	Maximum reporting delay	0	CrPsRepDelay_t	16
0xb	servUser	The default user of the service is the ground.	0	CrFwDestSrc_t	8
0xc	isSubSchedEnabled	Enable status of a sub-schedule	0	CrFwBool_t[SCD_N_SUB_TBS]	8

DPID	Name	Description	Default	Type	Size
0xd	isTbsEnabled	Enable status of time-based schedule	0	CrFwBool_t	8
0xe	nOfTbaInGroup	Number of TBAs in group	0	CrPsNTba_t[SCD_N_GROUP]	16
0xf	nOfTbaInSubSched	Number of TBAs in sub-schedule	0	CrPsNTba_t[SCD_N_SUB_TBS]	16
0x10	timeMargin	Time margin for time-based scheduling service	0	CrFwTime_t	32
0x11	AreYouAliveTimeOut	Time-out for the Are-You-Alive Test initiated in response to an On-Board Connection Test	0	CrFwTime_t	32
0x12	OnBoardConnectDestLst	Identifiers of target applications for an On-Board-Connection Test	0	CrFwDestSrc_t[TST_N_DEST]	32

Table 6.2: Datapool Variables

DPID	Name	Description	Default	Type	Size
0x13	lastEvtEid	Event identifier of the last generated level event report (one element for each severity level)	0	CrPsEvtId_t[4]	64
0x14	lastEvtTime	Time when last event report was generated (one element for each severity level)	0	CrFwTime_t[4]	128
0x15	nOfDetectedEvs	Number of detected occurrences of events (one element for each severity level)	0	CrPsNEvtRep_t[4]	64
0x16	nOfDisabledEid	Number of disabled event identifiers (one element for each severity level)	0	CrPsNEvtId_t[4]	64
0x17	nOfGenEvtRep	Number of generated event reports (one element for each severity level)	0	CrPsNEvtRep_t[4]	64
0x18	nOfAllocatedInCmd	Number of currently allocated InCommands (i.e. successfully created by the InFactory and not yet released)	0	CrPsNCmdRep_t	32
0x19	nOfAllocatedInRep	Number of currently allocated InReports (i.e. successfully created by the InFactory and not yet released)	0	CrPsNCmdRep_t	32
0x1a	nOfAllocatedOutCmp	Number of currently allocated OutComponents (i.e. successfully created by the OutFactory and not yet released)	0	CrPsNCmdRep_t	32
0x1b	nOfFailedInCmd	Number of InCommands whose creation by the InFactory failed	0	CrPsNCmdRep_t	32

DPID	Name	Description	Default	Type	Size
0x1c	nOfFailedInRep	Number of InReports whose creation by the InFactory failed	0	CrPsNCmdRep_t	32
0x1d	nOfFailedOutCmp	Number of OutComponents whose creation by the OutFactory failed	0	CrPsNCmdRep_t	32
0x1e	nOfTotAllocatedInCmd	Number of InCommands successfully created by the InFactory since application start	0	CrPsNCmdRep_t	32
0x1f	nOfTotAllocatedInRep	Number of InReports successfully created by the InFactory since application start	0	CrPsNCmdRep_t	32
0x20	nOfTotAllocatedOutCmp	Number of OutComponents successfully created by the InFactory since application start	0	CrPsNCmdRep_t	32
0x21	cycleCnt	Cycle Counter for Reports in RDL	0	CrPsCycleCnt_t[HK_N_REP_DEF]	64
0x22	debugVar	Value of Debug Variables	0	CrPsThirtytwoBit_t[HK_N_DEBUG_VAR]	96
0x23	sampleBufId	The i-th element of this array is the identifier of the Sampling Buffer for the i-th Report Definition in the RDL	0	CrPsSampleBufId_t[HK_N_REP_DEF]	32
0x24	lptRemSize	Remaining size of a large packet in the LPT Buffer (part of the large packet not yet down-transferred)	0	CrPsSize_t[LPT_N_BUF]	16
0x25	lptSize	Size of large packet in the LPT Buffer	0	CrPsSize_t[LPT_N_BUF]	16
0x26	lptSrc	Source of the large packet up-transfer to the LPT Buffer	0	CrFwDestSrc_t[LPT_N_BUF]	8

DPID	Name	Description	Default	Type	Size
0x27	lptTime	Time when last up-transfer command to the LPT Buffer was received	0	CrPsTimeSec_t[LPT_N_BUF]	32
0x28	nOfDownlinks	Number of on-going down-transfers	0	CrPsNOOfLinks_t	8
0x29	nOfUplinks	Number of on-going up-transfers	0	CrPsNOOfLinks_t	8
0x2a	partSeqNmb	Part sequence number for the up-/down/transfer to/from the LPT Buffer	0	CrPsPartSeqNmb_t[LPT_N_BUF]	16
0x2b	ctlRepDelay	Maximum reporting delay for the CTL in multiples of MON_PER	0	CrPsRepDelay_t	16
0x2c	ctlTimeFirstEntry	Time when first entry has been added to the CTL	0	CrFwTime_t	32
0x2d	funcMonEnbStatus	Functional monitoring enable status	0	CrPsEnDis_t	8
0x2e	nmbAvailFuncMon	Number of available functional monitors in the FMDL	0	CrPsFuncMonId_t	8
0x2f	nmbAvailParMon	Number of available parameter monitors in the PMDL	0	CrPsParMonId_t	16
0x30	nmbEnbFuncMon	Number of enabled functional monitors in the FMDL	0	CrPsFuncMonId_t	8
0x31	nmbEnbParMon	Number of enabled parameter monitors in the PMDL	0	CrPsParMonId_t	16
0x32	parMonEnbStatus	Enable state of parameter monitoring function	ENABLED	CrPsEnDis_t	8
0x33	firstTba	Identifier of next time-based activity due for release	0	CrPsNTba_t	16
0x34	isGroupEnabled	Enabled flag for time-based schedule group	1	CrFwBool_t[SCD_N_GROUP]	8

DPID	Name	Description	Default	Type	Size
0x35	isGroupInUse	InUse flag for time-based schedule group	0	CrFwBool_t[SCD_N_GROUP]	8
0x36	nOfGroup	Number of non-empty groups	0	CrPsNSubSchedGroup_t	8
0x37	nOfSubSched	Number of non-empty sub-schedules	0	CrPsNSubSchedGroup_t	8
0x38	nOfTba	Number of currently defined time-based activities (TBAs)	0	CrPsNTba_t	16
0x39	AreYouAliveSrc	Source of the latest (17,2) report received in response to a (17,1) command triggered by a (17,3) command	0	CrFwDestSrc_t	8
0x3a	AreYouAliveStart	Time when the Are-You-Alive Test is started in response to an On-Board Connection Test	0	CrFwTime_t	32
0x3b	OnBoardConnectDest	Destination of the (17,1) triggered by a (17,3) command	0	CrFwDestSrc_t	8
0x3c	failCode	Verification Failure Code	0	CrPsFailCode_t	8
0x3d	failCodeAccFailed	Failure code of last command which failed its Acceptance	0	CrPsFailCode_t	8
0x3e	failCodePrgrFailed	Failure code of last command which failed its Progress Check	0	CrPsFailCode_t	8
0x3f	failCodeStartFailed	Failure code of last command which failed its Start Check	0	CrPsFailCode_t	8
0x40	failCodeTermFailed	Failure code of last command which failed its Termination	0	CrPsFailCode_t	8
0x41	failData	Verification Failure Data (data item of fixed size but variable meaning)	0	CrPsFailData_t	32

DPID	Name	Description	Default	Type	Size
0x42	invDestRerouting	Destination of last command for which re-routing failed	0	CrFwDestSrc_t	8
0x43	nOfAccFailed	Number of commands which have failed their Acceptance Check	0	CrPsNOfCmd_t	16
0x44	nOfPrgrFailed	Number of commands which have failed their Progress Action	0	CrPsNOfCmd_t	16
0x45	nOfReroutingFailed	Number of commands for which re-routing failed	0	CrPsNOfCmd_t	16
0x46	nOfStartFailed	Number of commands which have failed their Start Action	0	CrPsNOfCmd_t	16
0x47	nOfTermFailed	Number of commands which have failed their Termination Action	0	CrPsNOfCmd_t	16
0x48	pcktIdAccFailed	Packet identifier of last command which failed its Acceptance Check	0	CrPsThirteenBit_t	13
0x49	pcktIdPrgrFailed	Packet identifier of last command which failed its Progress Action	0	CrPsThirteenBit_t	13
0x4a	pcktIdReroutingFailed	Packet identifier of last command for which re-routing failed	0	CrPsThirteenBit_t	13
0x4b	pcktIdStartFailed	Packet identifier of last command which failed its Start Check	0	CrPsThirteenBit_t	13
0x4c	pcktIdTermFailed	Packet identifier of last command which failed its Termination	0	CrPsThirteenBit_t	13
0x4d	stepPrgrFailed	Step identifier of last command which failed its Progress Check	0	CrFwProgressStepId_t	16

Table 6.3: Constants in the PUS Extension of the CORDET Framework

Name	Description	Value
HK_N_REP_DEF	Number of Report Definitions in the Report Definition List (maximum number of housekeeping/diagnostic reports which may be defined at any given time)	4
HK_MAX_SID	Maximum value of a service 3 Structure Identifier (SID)	10
HK_MAX_N_ITEMS	Maximum number of data items in a housekeeping/diagnostic report	4
HK_COLLECT_PER	Minimum collection period for service 3 reports	1
HK_MAX_N_SIMPLE	Maximum number of simply-commutated parameters in a housekeeping or diagnostic report	4
HK_MAX_N_GR	Maximum number of super-commutated groups in a housekeeping/diagnostic report	1
HK_MAX_REP	Maximum value of the repetition number of a super-commutated group in a housekeeping/diagnostic report	1
HK_MAX_N_REP	Maximum number of data items in a super-commutated groups in a housekeeping/diagnostic report	1
HK_N_SAMP_BUF	Number of service 3 Sampling Buffers	1
HK_N_DEBUG_VAR	Number of debug variables	3
TST_N_DEST	Number of destinations managed by test service	4
LPT_N_BUF	Number of Large Packet Transfer Buffers available for down-or up-transfer of large packets	1
MON_N_PMON	Maximum number of entries in the Parameter Monitoring Definition List PMDL (maximum number of parameter monitors in the application)	1
MON_N_LIM	Maximum number of parameter monitors with a limit check	1
MON_N_EXP	Maximum number of parameter monitors with an expected value check	1
MON_N_DEL	Maximum number of parameter monitors with a delta value check	1
MON_PER	Minimum monitoring period	1
MON_N_CLST	Maximum number of entries in the Check Transition List	1
MON_N_FMON	Maximum number of entries in the Functional Monitoring Definition List FMDL (maximum number of functional monitors in the application)	1
MON_N_PFMON	Maximum number of parameter monitors in a functional monitor	1
MON_NFPMON	Maximum number of functional monitors to which the same parameter monitor may belong	1

Name	Description	Value
MON_FPER	Period of execution of the Functional Monitoring Procedure expressed as a multiple of MON_PER	1
SCD_N_TBA	Number of time-based scheduled activities	1
SCD_N_SUB_TBS	Number of time-based sub-schedules	1
SCD_N_GROUP	Number of time-based schedule groups	1
EVT_MAX_N5S8	Maximum number of (5,8) reports which may be triggered in response to a single (5,7) command	2
EVT_DEST	Destination for out-going event reports	2
HK_DEST	Default destination of housekeeping reports	0

7 Type Definition

The table in this section defines the data types used by the PUS Extension of the CORDET Framework. For enumerated types, the list of enumerated values and their description is given.

Table 7.1: Types

Name	Description	Value
CrPsFailData_t	Type used for the Failure Data of a packet.	unsigned int
CrPsOneBit_t	Generic 1-bit type (least significant bit in 8-bit byte)	unsigned char
CrPsTwoBit_t	Generic 2-bit type (least significant 2 bits in 8-bit byte)	unsigned char
CrPsThreeBit_t	Generic 3-bit type (least significant 3 bits in 8-bit byte)	unsigned char
CrPsFourBit_t	Generic 4-bit type (least significant 4 bits in 16-bit word)	unsigned char
CrPsFourteenBit_t	Generic 14-bit type (least significant 14 bits in 16-bit word)	unsigned short
CrPsElevenBit_t	Generic 11-bit type (least-significants 11 bits in 16-bit word)	unsigned short
CrPsPartSeqNmb_t	Type for part sequence number	unsigned short
CrPsTransId_t	Type for transaction identifier	unsigned short
CrPsSID_t	Type for structure identifier	unsigned short
SID_N_OF_EVT	SID for HK packet holding number of generated events of each severity level	1
SID_HK_CNT	SID for HK packet holding the cycle counters for the HK packets	2
CrPsRepNum_t	Type used for repetition number	unsigned char
CrPsFailReason_t	Type for the transfer failure reason	unsigned short
CrPsNParMon_t	Type for the number of service 12 parameter monitors	unsigned char
CrPsNFuncMon_t	Type for the number of service 12 functional monitors	unsigned char
CrPsFuncMonCheckStatus_t	Type for the checking status of a service 12 functional monitor	unsigned char
MON_UNCHECKED	Functional monitor has not yet been notified since it was last enabled	1
MON_INVALID	The validity condition for the functional monitor is not satisfied	2
MON_RUNNING	The number of parameter monitors in the functional monitor which reported a monitoring violation is below the minimum failing number	3

Name	Description	Value
MON_FAILED	The number of parameter monitors in the functional monitor which reported a monitoring violation is greater than or equal to the minimum failing number	4
CrPsParMonCheckStatus_t	Type for the checking status of a service 12 parameter monitor	unsigned char
MON_VALID	Parameter is valid	1
MON_NOT_EXP	Parameter does not have the expected value	2
MON_ABOVE	Parameter value is above its upper limit	3
MON_BELOW	Parameter value is below its lower limit	4
MON_DEL_ABOVE	Parameter delta-value (difference between successive values) is above its upper limit	5
MON_DEL_BELOW	Parameter delta-value (difference between successive values) is below its lower limit	6
CrPsMonCheckType_t	Type of service 12 monitoring check	unsigned char
EXP_VAL_CHECK	Expected value check	1
LIM_CHECK	Limit check	2
DEL_CHECK	Delta check	3
CrPsEvtId_t	Type for Event Identifiers	unsigned short
EVT_DOWN_ABORT	Generated by an LPT State Machine when a down-transfer is aborted	1
EVT_UP_ABORT	Generated by an LPT State Machine when an up-transfer is aborted	2
EVT_MON_LIM_R	Generated when a Limit Check Monitoring Procedure has detected an invalid parameter value of real type	3
EVT_MON_LIM_I	Generated when a Limit Check Monitoring Procedure has detected an invalid parameter value of integer type	4
EVT_MON_EXP	Generated when a Expected Value Monitoring Procedure has detected an invalid parameter value of integer type	5
EVT_MON_DEL_R	Generated when a Delta Check Monitoring Procedure has detected an invalid parameter value of real type	6
EVT_MON_DEL_I	Generated when a Delta Check Monitoring Procedure has detected an invalid parameter value of integer type	7
EVT_FMON_FAIL	Generated when a functional monitor has declared a failure	8

Name	Description	Value
EVT_CLST_FULL	Generated when the Monitoring Function Procedure tries to add an entry to the Check Transition List but the list is full	9
EVT_DUMMY_1	Dummy level 1 event used for testing purposes	252
EVT_DUMMY_2	Dummy level 2 event used for testing purposes	253
EVT_DUMMY_3	Dummy level 3 event used for testing purposes	254
EVT_DUMMY_4	Dummy level 4 event used for testing purposes	255
CrPsValCheckExpVal_t	Type used for the expected value of a validity check	unsigned short
CrPsRepNumber_t	Type used for the repetition number of a service 12 parameter monitor	unsigned char
CrPsMinFailNmb_t	Type used for the minimum fail number of a service 12 functional monitor	unsigned char
CrPsParMonId_t	Type used for the identifier of a service 12 parameter monitor	unsigned short
CrPsFuncMonId_t	Type used for the identifier of a service 12 functional monitor	unsigned char
CrPsProtStatus_t	Type for the protected status of a service 12 functional monitor	unsigned char
UNPROTECTED	Not protected	0
PROTECTED	Protected	1
CrPsRepDelay_t	Type for the reporting of service 12 monitoring violations	unsigned short
CrPsAckFlag_t	Acknowledge Flag (least significant bit in 8-bit byte)	unsigned char
NO_ACK	No acknowledge required	0
ACK	Acknowledge required	1
CrPsThirteenBit_t	Generic 13-bit type (least significant 13 bits in 16-bit word)	unsigned short
CrPsSixteenBit_t	Generic 16-bit type	unsigned short
CrPsThirtytwoBit_t	Generic 32-bit Type	unsigned int
CrPsPrgStep_t	Type for TC Progress Step	unsigned char
CrPsNPar_t	Type for number of parameters	unsigned char
CrPsParId_t	Type used for parameter identifier	unsigned short
CrPsNGroups_t	Type for number of groups	unsigned char
CrPsNSID_t	Type for the number of SIDs	unsigned char
CrPsEnDis_t	Generic enabled/disabled type	unsigned char

Name	Description	Value
DISABLED	Disabled	0
ENABLED	Enabled	1
CrPsNEvtId_t	Type for number of event identifiers	unsigned short
CrPsEightBit_t	Generic 8-bit type	unsigned char
CrPsNEvtRep_t	Type for number of event reports	unsigned short
CrPsCycleCnt_t	Type for the cycle counter and HK report periods	unsigned short
CrPsSampleBufId_t	Type for identifiers of the sampling buffer	unsigned char
CrPsTimeSec_t	Type for a real-valued time expressed in seconds	unsigned int
CrPsNOOfCmd_t	Type used for the number of commands	unsigned short
CrPsNOOfLinks_t	Type for number up- and down-links	unsigned char
CrPsSize_t	Type for the size of a large packets	unsigned short
CrPsNCmdRep_t	Type for number of commands and reports in component factories	unsigned int
CrPsNTba_t	Type for number of time-based scheduled activity (TBA)	unsigned short
CrPsNSubSchedGroup_t	Type for the number of sub-schedules and of groups	unsigned char
CrPsFailCode_t	Type used for a service 1 Failure Code	CrFwOutcome_t
VER_CMD_INV_DEST	Failure code for all (1,10) reports	129
VER_REP_CR_FD	Failure code for start actions when they unsuccessfully attempt to create a new report from the OutFactory	130
VER_OUTLOADER_FD	Failure code for start actions when the Load operation in the OutLoader has failed	131
VER_SID_IN_USE	A (3,1) or (3,2) command attempted to create a new report with a SID which is already in use	132
VER_FULL_RDL	A (3,1) or (3,2) command attempted to create a new report at a time when the RDL is already full	133
VER_ILL_DI_ID	A service 3 command carried an illegal data item identifier	134
VER_ILL_NID	A service 3 command carried too many data item identifiers	135
VER_ILL_SID	A service 3 command had an invalid SID	136
VER_ENB_SID	A service 3 command encountered an enabled SID	137

Name	Description	Value
VER_MI_S3_FD	A multi-instruction service 3 command has failed	138
VER_FACT_PRGR_FD	The progress action of a multi-instruction service 3 command has failed to retrieve a report from the OutFactory	139
VER_ILL_EID	The start action of a service 5 command has encountered an illegal Event Identifier (EID)	140
VER_EID_ST_FD	All the instructions in a service 5 command have been rejected	141
VER_ILL_MON	A Parameter or Functional Monitor Identifier in a service 12 command is out-of-range or not defined	142
VER_MON_START_FD	All the instructions in a service 12 command have been rejected	143
VER_PMDL_FULL	A service 12 command has found the Parameter Monitor Definition List (PMDL) full	144
VER_MON_ILL_DI	A service 12 command has found the data item identifier of the parameter to be monitored illegal	145
VER_MON_PROT	A service 12 command as found a parameter monitor which belongs to a protected functional monitor	146
VER_MON_ENB	A service 12 command has found a parameter or functional monitor which is enabled	147
VER_MON_USE	A service 12 command has found a parameter monitor which is used by a functional monitor	148
VER_FMDL_FULL	A service 12 command has found a Functional Monitor Definition List (FMDL) full	149
VER_MON_TMP	A service 12 command has found too many parameter monitors in a functional monitor	150
VER_MON_MFN	A service 12 command has found a value of minimum failing number equal to zero	152
VER_SCD_ILL_SS	Failure code for start action of service 11 command when it finds an illegal sub-schedule identifier	153
VER_FULL_TBS	A service 11 command found the Time-Based Schedule (TBS) full	154
VER_SCD_ILL_G	A service 11 command found an illegal schedule group identifier	155
VER_SCD_ILL_RT	A service 11 command found an illegal release time	156
VER_SCD_ILL_DS	A service 11 command found an illegal destination for an scheduled command	157
VER_SCD_CRFAIL	A service 11 command was unable to create an InCommand for a scheduled command (either due to lack of resources or due to illegal command type)	158

Name	Description	Value
VER_SCD_ST_FD	All instructions in a service 11 command have been rejected	159
VER_ILL_ACT_ID	Command (11,5) was unable to find an activity identifier in the TBS	160
VER_TST_TO	The time-out of the (17,3) command has triggered	161
VER_CRE_FD	The InLoader has failed to create an InCommand to hold an incoming command	254
VER_CMD_LD_FD	The InLoader has failed to load an InCommand component into its InManager	255