TECNOSPAZIO S.p.A   DOC. N°:SD2-SUM-TS-011
Date: October 2002   Revision : E
Project: SHARK   Sheet N°: 1

Title:

SD2 Subsystem

User Manual

Project: SHARK- ROSETTA LANDER

Contract: ASI-TS- 67/I.o./97 (WP 4550)

		Function	Name	Signature
Prepared	By :	SW development	P.Bologna	
	By :	SW development	E.Crudo	
Checked	By :	Project Leader	P.G. Magnani	
	By :	SW development	C.Fantinati	
Approved	By :	P.A. Manager	M. Colomba	
	By :	Program Manager	E. Re	
Authorised	By :	General Manager	L.Giardino	

Configuration
Management :   Rev.: E   Date:

Accepted by :   _____   _____

CHANGE HISTORY RECORD

Rev.	Date	Change description	O.M.	Prepared / modified by
-		first issue	-	P.Bologna
A		Updated for FM delivery	819	P.Bologna
B		Closure of action A5a-TS specified in RO-LAN-SD2-MOM-04/01 SD2-PSR/DRB Part 2	821	G.Soavi
C		Completed the sections relevant to SD2 specific commands, error messages, SD2 operations. Included also the tables mapping speed level and torque level to RPM and mA	825	P.Bologna
D		Added the table 6- oven positions	860	P.Bologna
E		Added GRM Constraints	867	E.Cudo

File name:	sum_e.doc

REVISION INDEX OF SHEETS

Sheets	Revision	Sheets	Revision	Sheets	Revision
1..33	-				
1..34	A				
1..36	B				
1..71; B1..B3; C1; D1; E1; F1	C				
1,2,3	D				
4,5	C				
6	D				
7..71; B1..B3; C1; D1; E1; F1	C				
G1	D				
1,2,3	E				
4,5	C				
6	E				
7-48	C				
49	E				
50-71; B1..B3; C1; D1; E1; F1	C				
G1	D				

TECNOSPAZIO S.p.A   DOC. N°:SD2-SUM-TS-011
Date: October 2002   Revision : E
Project: SHARK   Sheet N°: 4

TECNOSPAZIO S.p.A   DOC. N°:SD2-SUM-TS-011
Date: October 2002   Revision : E
Project: SHARK   Sheet N°: 5

DISTRIBUTION LIST

Internal Distribution

Name	Copies
P.G. Magnani	1
E. Re	1

External Distribution

Name	Company	Copies
R. Mugnuolo	Asi- Matera	1
A. Olivieri	Asi- Matera	1

LIST OF CONTENTS

1. INTRODUCTION

1.1 INTENDED READERSHIP

The readers of this document are intended to be the members of SD2 operation and maintenance team

1.2 APPLICABILITY STATEMENT

This SUM applies to SD2 SW, CI SD2C.

1.3 PURPOSE OF THE DOCUMENT

Purpose of this document is to explain how to command SD2 and how to interpret the telemetry data generated by SD2.

1.4 HOW TO USE THIS DOCUMENT

Chapter 2 reports a general overview of SD2 subsystem. It shall be read as first.

Chapter 3 contains:

- a description of SD2 states, with transition diagram; each SD2 status is described by means of a functional description, cautions and warnings, cross reference to other operations

- a description of the SD2 specific commands; each command is detailed by means of a functional description, caution and warnings, cross reference to telemetry fields

- a description of all the checks performed by SD2; each check is detailed by means of a functional description, caution and warnings, trouble-shooting

- a description of the error handling. It is reported the kind of errors detected by SD2 SW, the kind of recovery actions executed, the way to disable/enable the recovery actions

- the table containing the description and the value of the constants hard coded in the software code

- a description of the timing used in telemetry data

Chapter 4 contains the constraints applied to SD2. These constraints shall be taken into account every time SD2 is used.

Chapter 5 reports the procedures for the locking, de-locking, for carousel and drill translation zeroing. These procedures shall be used only for maintenance reason.

Chapter 6 reports an example of SD2 operation made of drilling, sample collection, carousel motion, sample discharge, volume checker motion, drill translation and carousel homing.

Appendix A contains a detailed description of error messages of SD2 software.

Pages B1..B3 reports the hexadecimal format of commands to power on/off the SD2 devices by means a direct access to SD2 electronics board registers.

Page C1 reports the mapping between speed level and torque level to RPM and mA values for drill translation motion.

Page D1 reports the mapping between speed level and torque level to RPM and mA values for carousel motion.

Page E1 reports the mapping between speed level and torque level to RPM and mA values for drill rotation motion.

Page F1 reports the mapping between speed level and torque level to RPM and mA values for volume checker motion.

1.5 RELATED DOCUMENTS

INTICD: SHARK-SA-TS-044

       SD2 Electronic Unit Internal Interface Control Document

CDMSICD: SHARK-ICD-TS-043

       CDMS- SD2 Data Interface Control Document

URD: SHARK-URD-TS-067

       SD2 Software User Requirements Document

LFT: SD2-IC-TS-010

       SD2 Subsystem Limited Functional Tests

SD2_QA: SHARK-QA-TS-005

       Product Assurance Plan for Shark Program

SDE: SD2C-IA-TS-003

       SD2 Software Development Environment

1.6 CONVENTIONS

Specific commands are written in <u>underlined</u>.

1.7 PROBLEM REPORTING INSTRUCTIONS

Problems are reported by means of Software Problem Report, as defined in [SD2_QA], Appendix A.

2. SD2 OVERVIEW SECTION

2.1 SD2 GENERAL OVERVIEW

The SD2 subsystem provides samples collected at different depths to microscopes and evolved gas analyzers.

Specifically SD2 is able to:

- drill the comet surface to a suited depth ranging from the surface down to about 230 mm;

- collect several samples of material on a stepwise way from pre-determined known depths;

- transport each sample to the Carousel sample feed position for delivering it to different ovens for high and medium temperatures and microscope inspection;

- rotate the Carousel and present the sample fed oven in front of the appropriate experiment station for analysis.

The SD2 Subsystem consists of the following components:

- Mechanical Unit

- Electronic Unit

- Software (SW)

- Harness

The **Mechanical Unit** is mounted at the Lander Balcony compartment in correspondence with a hole in the Balcony dedicated to the drilling-sampling operations and will perform all electromechanical functions.

The unit consists of the following main components: Tool Box, Carousel, Volume Checker, Base Plate.

The Tool Box contains the drill-sampler tool in a protective structural shell (in composite material) which assures that no external contamination can reach the tools and the actuators inside.

The drilling and sampling functions are integrated in a unique auger.

During the final phase of the drilling operation, the sampling mechanism collects the sample. The sample collected is then delivered to the dedicated containers (ovens) allocated on the rotating disk (Carousel).

A rotation actuator enables the Carousel to mate each oven with the experiment interface positions and the sample feed position.

The Volume Checker measures the volume of the material discharged into the oven.

The Base Plate is a plate where the Carousel and the Tool Box are mounted on.

It provides I/F for Çiva-M and for the Tapping Stations.

The **Electronic Unit** is installed into the warm compartment of the Lander and incorporates all electronics to control the Mechanical Unit. It provides a HW platform to run the **SW** which implements SD2 functions under high level control system (CDMS) commands. The unit incorporates the Com-DPU processor board developed by DLR for the Lander.

The **Harness** electrically connects Mechanical and Electronic Units as well as SD2 to Lander's Common E-Box.

Accommodation of the SD2 units on the Lander is outlined in the figure 2.1-1.

TECNOSPAZIO S.p.A   DOC. N°:SD2-SUM-TS-011
Date: October 2002   Revision : E
Project: SHARK   Sheet N°: 12

MECHANICAL UNIT

ELECTRONIC UNIT

CAROUSEL
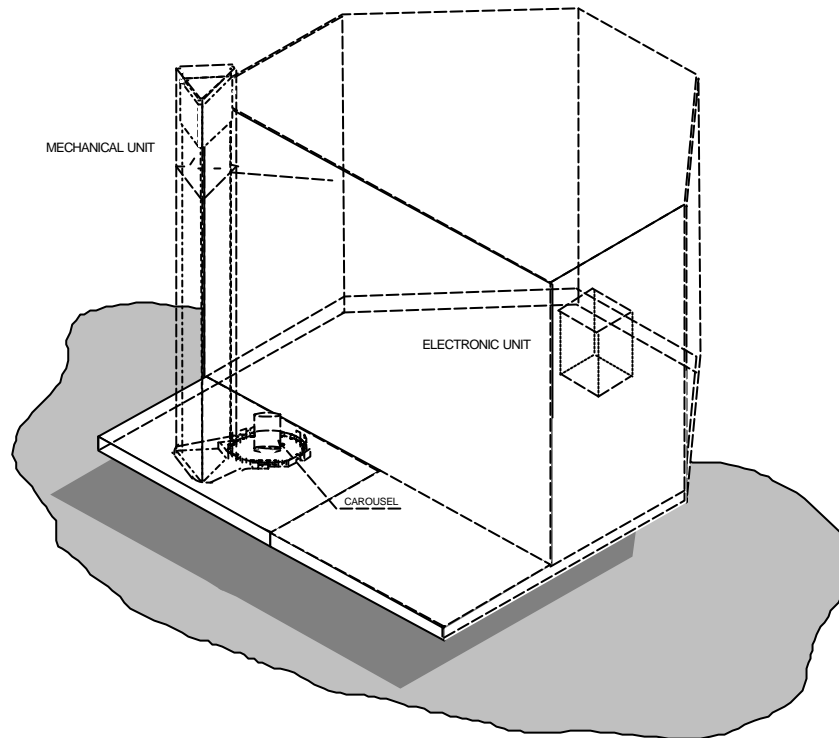
Figure 2.1-1   SD2 units on Lander

2.2 SD2 SOFTWARE PURPOSE

The main functions of SD2 SW are the following:

- receive and execute commands from the Command and Data Management System (CDMS). From CDMS, SD2 SW can receive *Standard Commands*, that can be sent by CDMS also to other units, and *Specific Commands*, i.e. command specific for the SD2 purpose

- generate telemetry data. The telemetry data consists of:

  - scientific data; they are autonomously collected by SD2 SW at the default frequency 4[Hz]. A specific command allows to change the acquisition frequency and to start the transmission to CDMS of collected data

  - housekeeping data; these data are automatically requested by CDMS to SD2 SW, each word every 2 [s]

  - backup RAM data; this data are written by SD2 SW to exchange data with the other units

- command the motion of mechanical units by sending proper commands to the Electronics Unit

- perform checks on the received commands, both syntax checks and interference checks

- perform cyclic checks on the commanded motion

- activate recovery procedures when a failure is detected.

According to a slave concept of SD2 subsystem and to its possible critic operations, there are only two automatic recovery actions performed by SD2:

- the recovery action related to drill translation speed check failure: this recovery action consists of two steps. First of all, when a failure in the drill speed check is detected, the drill translation motion continues by using the recovery torque level instead of the commanded torque level. If the speed check fails again, then the drill translation motion continues by using the recovery torque level and both the main and the redundant windings. If the check fails again, then SD2 enters in emergency status

- when a failure different to drill translation speed check failure is detected, SD2 SW starts automatic execution of either emergency-stop, or halt, or warning, or ignore procedure, according to the failure severity

2.3 SD2 SOFTWARE INTERFACES

The SD2 SW interfaces are:

- the CDMS, through the ComDPU. This interface is used to receive commands and to send telemetry data.

- the CDMS backup RAM of COSAC, PTOLEMY and LANDG. This interface is used to acquire the status of COSAC, PTOLEMY, LANDG in order to check if the motion of SD2 mechanical units is allowed

- the CDMS backup RAM of SD2 itself (DRILL backup RAM). This interface is used to communicate to other units the SD2 status

- the Electronics Unit. This interface is used to send commands in order to move the Mechanical Unit.

The CDMS interface is specified in [CDMSICD].

The Electronics interface is specified in [INTICD].

3. SD2 REFERENCE SECTION

3.1 SD2 STATES

The SD2 can be in one out of the following states:

- undefined status

- ready

- dead

- drill in progress

- drill completed

- sampling in progress

- sampling completed

- carousel motion in progress

- carousel motion completed

The SD2 status is notified to CDMS:

- in the word #13 called OPST of housekeeping data

- in the word # 17 called OPST of scientific data

- in the word # 1 called "SD2 current status" of SD2 backup RAM

as described in [CDMSICD]. The updated of SD2 backup RAM is performed only when the SD2 status changes.

The following transition diagram applies to the SD2 states:
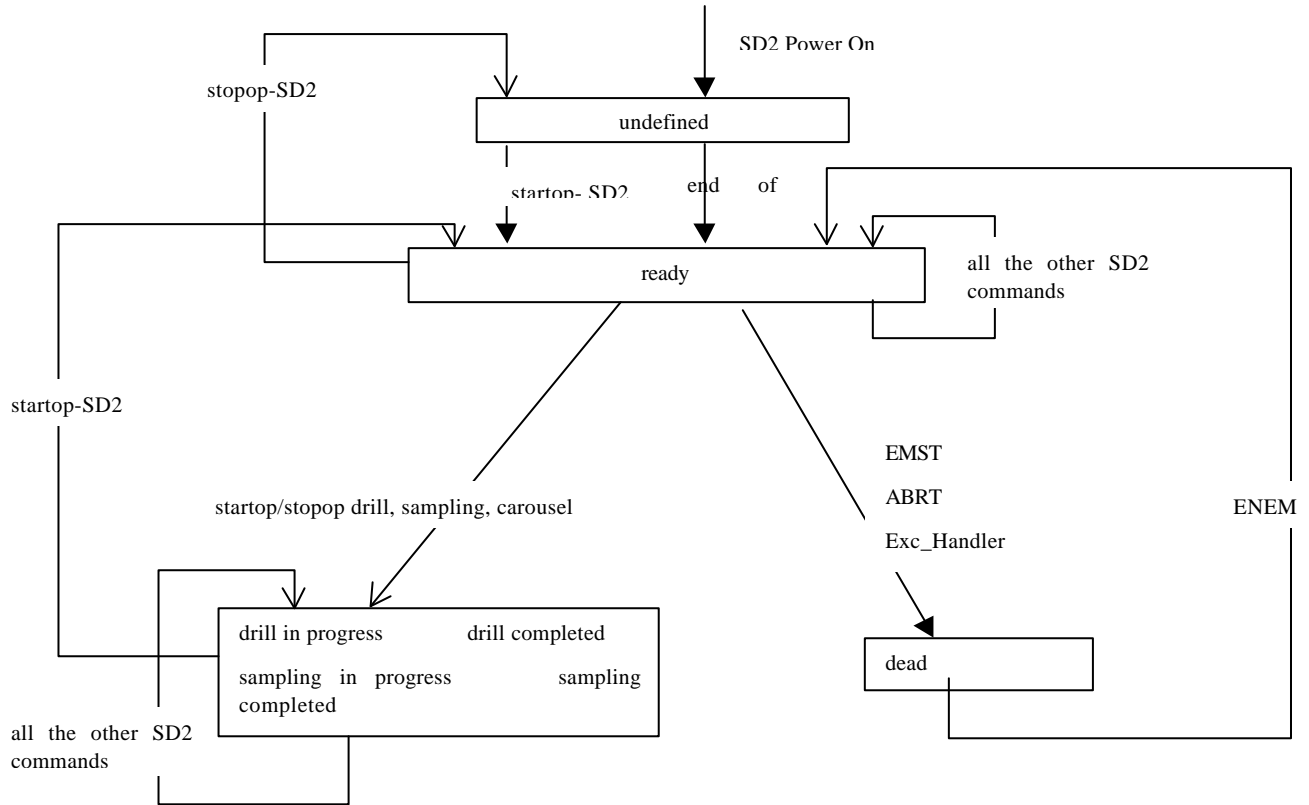
Fig. 3.1-1. Transition diagram relevant to SD2 states

3.1.1 SD2 Undefined Status

Functional Description

This is the status in which no SD2 activities is in progress: all the motor drivers are powered OFF, no speed-trajectory generation is in progress, no scientific data dump is in progress, no request versus CDMS is in progress. Only the following activities continue:

- acquisition of scientific data

- reception of commands from CDMS

- transmission of housekeeping data, obviously if required by CDMS.

When in undefined status, SD2 can be safely powered OFF.

Cautions and warnings

- When in "undefined status", SD2 SW rejects any specific command with exception of the specific command <u>STARTOP</u> with parameter "SD2-operation".

- SD2 is in "undefined" status:

- during SD2 initialization

- when the command <u>STOPOP</u> with parameter "SD2 operation" is received

- The "undefined" status can be changed:

- autonomously by SD2 SW itself, at the end of its successful initialization

- when the specific command <u>STARTOP</u> with parameter "SD2-operation" is received; it allows to go to "ready" status

Cross reference to other operations

See <u>STARTOP</u> <u>STOPOP</u> commands

3.1.2 SD2 Ready Status

Functional Description

This is the status in which SD2 accepts all the specific commands and can performs all its activities: acquisition of scientific data, dumping of scientific data, sending requests to CDMS, control motion of mechanical devices, reception of new CDMS commands, execution of cyclic checks.

Cautions and warnings

- SD2 is in "ready" status:

- at the end of SD2 SW initialization

- when the command <u>STARTOP</u> with parameter "SD2 operation" is received

- when the command <u>ENEM</u> is received

- The "ready status" can be changed by means of the specific commands <u>STARTOP, STOPOP</u>, <u>ABRT</u>, <u>EMST, ENEM</u> and by the recovery actions executed by SD2 SW.

Cross reference to other operations

See <u>STARTOP</u> <u>STOPOP</u> <u>ABRT</u> <u>EMST</u> <u>ENEM</u> commands

3.1.3 SD2 Dead Status

Functional Description

This is the status in which SD2 enters either because the specific commands <u>EMST</u> or <u>ABRT</u> are received, or because a failure is detected by the SD2 SW and the related recovery action causes to enter in "dead" status.

Cautions and warnings

- The "dead" status is a non-nominal SD2 status, meaning that something was not working properly. For safety reason, when in "dead" status the SD2 SW removes all the specific commands currently present in the command queue, such as the specific commands of the mission plan under execution. So, *when in "dead" status, it is required a manual intervention to go on the mission execution*.

Moreover, when in "dead" status, SD2 rejects any specific command with exception of <u>ENEM</u>, <u>MHIT</u>, <u>STOPOP</u> with parameter "SD2 Operation". <u>STOPOP</u> with parameter "SD2 Operation" allows entering in "undefined" status for a safety SD2 shutdown. <u>MHIT</u> allows dumping the acquired scientific data, so that it is possible to know the reason of the failure. <u>ENEM</u> allows entering in "ready" status and so continuing the nominal SD2 processing. As stated above, these specific commands can only be sent by a manual intervention.

- The "dead" status can be changed only by means of the specific command <u>ENEM</u>

Cross reference to other operations

See <u>ABRT</u> <u>EMST</u> commands and "SD2 Recovery Actions" chapter

3.1.4 SD2 Drill in Progress Status

Functional Description

This is the status in which SD2 enters only because the specific command <u>STARTOP</u> with parameter "drilling" is received. This status is functionally the same as the "ready" status. It has been defined because other Units require this information.

Cautions and warnings

The "drill in progress" status is functionally the same as the "ready" status.

The "drill in progress" status can be changed only by means of the specific commands <u>STARTOP</u> <u>STOPOP.</u>

Cross reference to other operations

See <u>STARTOP</u> <u>STOPOP</u> commands

3.1.5 SD2 Drill Completed Status

Functional Description

This is the status in which SD2 enters only because the specific command <u>STOPOP</u> with parameter "drilling" is received. This status is functionally the same as the "ready" status. It has been defined because other Units require this information.

Cautions and warnings

The "drill completed" status is functionally the same as the "ready" status.

The "drill completed" status can be changed only by means of the specific commands <u>STARTOP</u> <u>STOPOP.</u>

Cross reference to other operations

See <u>STARTOP</u> <u>STOPOP</u> commands

3.1.6 SD2 Sampling in Progress Status

Functional Description

This is the status in which SD2 enters only because the specific command <u>STARTOP</u> with parameter "sampling" is received. This status is functionally the same as the "ready" status. It has been defined because other Units require this information.

Cautions and warnings

The "sampling in progress" status is functionally the same as the "ready" status.

The "sampling in progress" status can be changed only by means of the specific commands <u>STARTOP</u> <u>STOPOP.</u>

Cross reference to other operations

See <u>STARTOP</u> <u>STOPOP</u> commands

3.1.7 SD2 Sampling Completed Status

Functional Description

This is the status in which SD2 enters only because the specific command <u>STOPOP</u> with parameter "sampling" is received. This status is functionally the same as the "ready" status. It has been defined because other Units require this information.

Cautions and warnings

The "sampling completed" status is functionally the same as the "ready" status.

The "sampling completed" status can be changed only by means of the specific commands <u>STARTOP</u> <u>STOPOP</u>.

Cross reference to other operations

See <u>STARTOP</u> <u>STOPOP</u> commands

3.1.8 SD2 Carousel in Progress Status

Functional Description

This is the status in which SD2 enters only because the specific command <u>STARTOP</u> with parameter "carousel" is received. This status is functionally the same as the "ready" status. It has been defined because other Units require this information.

Cautions and warnings

The "carousel in progress" status is functionally the same as the "ready" status.

The "carousel in progress" status can be changed only by means of the specific commands <u>STARTOP</u> <u>STOPOP.</u>

Cross reference to other operations

See <u>STARTOP</u> <u>STOPOP</u> commands

3.1.7 SD2 Carousel Completed Status

Functional Description

This is the status in which SD2 enters only because the specific command <u>STOPOP</u> with parameter "carousel" is received. This status is functionally the same as the "ready" status. It has been defined because other Units require this information.

Cautions and warnings

The "carousel completed" status is functionally the same as the "ready" status.

The "carousel completed" status can be changed only by means of the specific commands <u>STARTOP</u> <u>STOPOP.</u>

Cross reference to other operations

See <u>STARTOP</u> <u>STOPOP</u> commands

3.2 SD2 SPECIFIC COMMANDS

The communication between SD2 and CDMS occurs exclusively on Action Code and Request Code basis. The commands contained in the Action Code are called *Standard Commands* and can be sent by CDMS to all the Rosetta instruments. By means of the Action Code RCMD, it can be sent to SD2 the commands specific to SD2, called *SD2 Specific Commands.*

The SD2 Specific Commands are split into two types: *Normal Specific Commands* and *Immediate Specific Commands.*

The SD2 Normal Specific Commands are stored in a RAM FIFO buffer whose dimension allows storing up to C_MAX_MP_WORDS words. The queue is processed sequentially, each specific command being processed when the execution of the previous one is successfully completed. The execution of a specific command is completed when the target specified by the command has been reached. So, for a controlled position command, the specific command execution is completed when the target position has been reached and the motion is stopped, while for a speed controlled command, the command execution is completed when the target speed is reached even if the motion continues.

The SD2 Immediate Specific Commands are executed exactly when they are received by SD2, despite of the queue of specific normal commands and despite of the status of the command currently under execution.

The SD2 specific commands are hereunder split in:

- carousel motion: <u>CAPO</u>, <u>CASI</u>

- drill translation motion: <u>DRTR</u>, <u>DRTT</u>, <u>DRTC</u>

- drill rotation motion: <u>DRGO</u>, <u>DRST</u>

- volume checker motion: <u>MVCK</u>, <u>VCAC</u>

- sampling tube activation: <u>SARE</u>

- emergency commands: <u>EMST</u>, <u>ABRT</u>, <u>ENEM</u>

- <u>WRAD</u> specific command: direct access to SD2 electronic board registers

- <u>RDAD</u> specific command: read SD2 electronic board register

- general driver commands: <u>ZERO</u>, <u>ONOF</u>, <u>ACRE</u>

- <u>LANDG</u> specific command: set parameters relevant to landing gear backup RAM

- commands for notifying SD2 current status and the end of operation: <u>STARTOP</u>, <u>STOPOP</u>

- <u>EHEN</u> specific command: enable/disable of recovery action related to a detected failure

- <u>MHIT</u>: scientific data management

- <u>DELAY</u>: wait for the specified time before starting the processing and the execution of next specific command.

Specific command formats are detailed described in [CDMSICD].

The numbering of scientific and housekeeping data words hereunder mentioned starts from 0.

The numbering of bits hereunder mentioned starts from 0.

Cross reference to other operations

The checks performed before/ during the command execution are specified in chapter "SD2 Checks". The recovery actions are specified in chapter "SD2 Recovery Actions". The errors raised during command execution are listed in Appendix A. For the value of constants, refer to chapter "SD2 SW Constants".

3.2.1 Carousel Motion

Functional Description

In order to move carousel, the commands <u>CAPO</u> (move carousel to the specified position) and <u>CASI</u> (move carousel to scientific port) can be used. Before the motion, the volume checker is moved up in order to guarantee no collision between the volume checker and the ovens.

The two carousel commands require specifying the speed and the torque level to be applied. At page D1 it is reported the mapping between these levels and the RPM mA values (column "speed level used by the speed handler"). See [CDMSICD] "SD2 Reference Systems" for oven and port positions.

When a carousel command is received, SD2 automatically powers on the carousel motor and the carousel resolver. Then it starts up a speed ramp with the specified torque level starting from speed level 0 up to the speed specified by the command, each speed level being generated at C_MRFR frequency. The command is completed when the position error (difference between the current and the target position) is less or equal to C_CAR_TARGET_TH.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

- tapping station status check

- SD2 mechanical correctness check

During the motion, SD2 SW performs the carousel speed check.

Moreover, SD2 SW checks that the ramp down phase is completed within a fixed timeout, otherwise the failure EC_PH_CAR_RAMP_DOWN_TIMEOUT with severity 3 is raised.

Cross reference to telemetry fields

- position of carousel: word#1 of scientific data; word#6 of housekeeping data

- carousel motor direction: bit 15 of word#7 of scientific data; bit 15 of word#7 of housekeeping data

- carousel driver status: bit 13 of word#8 of scientific data; bit 13 of word#8 of housekeeping data

- carousel resolver status: bit 9 of word#8 of scientific data; bit 9 of word#8 of housekeeping data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.2 Drill translation motion

Functional Description

In order to move drill translation, the commands <u>DRTR</u> (drill translation), <u>DRTT</u> (drill translation with timeout), <u>DRTC</u> (Drill translation for driver check) can be used.

The drill translation commands require specifying the speed and the torque level to be applied. At page C1 it is reported the mapping between these levels and the RPM mA values (column "speed level used by the speed handler").

When a drill translation command is received, SD2 automatically powers on the drill translation resolver and the drill translation motor. Then it starts up a speed ramp with the specified torque level starting from speed level 0 up to the speed specified by the command, each speed level being generated at C_MRFR frequency. The command is completed as soon as the current position oversteps the target position.

Cautions and warnings

Before starting a drill translation motion, the <u>LANDG</u> command defining the lander leg forbidden ranges shall be sent to SD2.

Before starting <u>DRTR</u> command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

- landing gear position check

- SD2 mechanical correctness

During the motion, SD2 SW performs the drill translation speed check.

Each drill translation motion starts by powering only main winding and by using the torque specified in the relevant specific command. If during the motion a failure in the drill translation speed check is detected, the recovery action related to error severity 2 is automatically executed (see chapter "SD2 Recovery Actions").

Before starting <u>DRTT</u> command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

- landing gear position check

- SD2 mechanical correctness check

During the motion, SD2 SW does not perform the drill translation speed check, and it terminates the command execution either because the target position has been reached, or because the time specified in the command elapsed. The use of this command is suggested for drill translation motion within an oven. In fact in this case the target position could not be reached because some material is present in the oven. After the command execution, the user is in charge to check if the reached position is OK.

Before starting <u>DRTC</u> command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

- landing gear position check

- SD2 mechanical correctness check

During the motion, SD2 SW does not perform the drill translation speed check. If the target position has not been reached within the timeout specified in the command, the failure EC_DRTR_MAIN_MOTION_CHECK_FAILURE or EC_DRTR_RED_MOTION_CHECK_FAILURE is raised,

according to the device specified in the command. The severity of these errors is 3, so that SD2 enters in DEAD status. The use of this command is suggested for checking the drill translation main and redundant windings before starting a drilling mission plan execution.

Examples

1. Before starting a drilling operation, it is useful to check if the drill translation is working. The next command sequence allows to perform the drill translation winding check:

- starting drill position: 0; starting carousel position: 0

- send the command DRTC 20 4 0 200 120 – drill translation with speed level 20 (563 RPM), torque 4 (375 mA), performed by powering on the main winding (parameter 0) down to the commanded position 200 [1/100mm], within 120 [250 ms]

- send the command DRTC 20 4 1 0 120 – drill translation with speed level 20 (563 RPM), torque 4 (375 mA), performed by powering on the redundant winding (parameter 1) up to the commanded position 0 [1/100mm], within 120 [250 ms].

If after the main winding motion the failure EC_DRTR_MAIN_MOTION_CHECK_FAILURE is raised, it means that the main winding does not work properly. If after the redundant winding motion the failure EC_DRTR_RED_MOTION_CHECK_FAILURE is raised, it means that the redundant winding does not work properly. If one of this failure raises, it means that a drilling operation could fail; perhaps it could happen that will not be possible to move up the drill, and in this situation will not be possible to move anymore the carousel and perhaps it could be some interference with balcony and with other Rosetta payloads. So, according to the result of the drill translation winding check, the user who is in charge to plan the Rosetta whole mission can select either to continue the SD2 mission, or to post-pone it after conclusion of mission of the other payloads.

Cross reference to telemetry fields

- position of drill: word#0 of scientific data; word#5 of housekeeping data

- drill translation motor direction: bit 13 of word#7 of scientific data; bit 13 of word#7 of housekeeping data

- drill translation driver status: bits 11 and 12 of word#8 of scientific data; bits 11 and 12 of word#8 of housekeeping data

- drill resolver status: bit 10 of word#8 of scientific data; bit 10 of word#8 of housekeeping data

- status of recovery action applied: bits 9,10,11 of word#7 of scientific data; bits 9,10,11 of word#7 of housekeeping data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.3 Drill Rotation motion

Functional Description

In order to command the drill rotation device, the commands <u>DRGO</u> and <u>DRST</u> can be used. The DRGO command allows starting the drill rotation motion, while <u>DRST</u> allows stopping the rotation.

The drill rotation command requires specifying the speed and the torque level to be applied. At page E1 it is reported the mapping between these levels and the RPM mA values (column "speed level used by the speed handler"). Moreover, it requires the time duration of the drill rotation motion.

TECNOSPAZIO S.p.A

Date: October 2002

Project: SHARK

DOC. N°:SD2-SUM-TS-011

Revision : E

Sheet N°: 23

When <u>DRGO</u> is received, SD2 automatically powers on the drill rotation motor. Then it starts up a speed ramp with the specified torque level starting from the current speed level (0 if the drill rotation was at rest) up to the speed specified by the command, each speed level being generated at C_MRFR frequency. When the time specified by <u>DRGO</u> is elapsed, or when the <u>DRST</u> command is received, SD2 starts a speed ramp starting from the current speed level up to speed level 0 and powers off the driver. When the coasting speed starts, SD2 activates the drill rotation counter and starts the drill rotation speed check. When the deceleration ramp starts, SD2 power off the drill rotation counter and stops the drill rotation speed check.

As far as the command processing is concerned, the execution of <u>DRGO</u> is completed when the speed specified by the command is reached. This means that:

- at the of speed ramp up, the drill rotation continues at coasting speed

- at the end of speed ramp up, the execution of another specific command can start. This allows for example to start the drill translation while the drill rotation is in progress

- at the end of speed ramp, other <u>DRGO</u> commands can be sent. Their effect is to start up a new speed ramp with the specified torque level starting from the current speed up to the speed specified by the new command. The drill rotation will stop either because a <u>DRST</u> is received, or because the time specified in the last <u>DRGO</u> command is elapsed

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

- SD2 mechanical correctness check

During the motion, SD2 SW performs the drill rotation speed check.

Cross reference to telemetry fields

- drill rotation measurement: word #3 of scientific data

- drill rotation motor direction: bit 14 of word#7 of scientific data; bit 14 of word#7 of housekeeping data

- drill rotation driver status: bit 15 of word#8 of scientific data; bit 15 of word#8 of housekeeping data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

Examples

1. The DRGO 31 0 0 10 commands a drill rotation with speed level 31 (1250 rpm), torque level 0 (250 mA), direction CW. The coasting speed is reached after 31*0.250=7.75 [s] and is maintained for 10 [s].

3.2.4 Volume Checker motion

Functional Description

In order to command the volume checker motion, the commands <u>MVCK</u> and <u>VCAC</u> can be used.

The <u>MVCK</u> command moves the volume checker in the direction specified in the command.

If the commanded direction of <u>MVCK</u> is DOWN, the motion is stopped when the time specified by the command elapses.

If the commanded direction of <u>MVCK</u> is UP, the motion is stopped as soon as the upper microswitch of the volume checker is reached. The "waiting time for volume checker movement completion" specified by the command is used as timeout for the motion, and it is used to perform the "Volume checker microswitch check".

At the end of <u>MVCK</u> command, the word # 2 VCD of scientific data is filled in.

The <u>VCAC</u> command executes automatically the following command sequence:

- move volume checker up with speed level and torque level specified by the command; at the end, the word # 2 VCD and the word # 4 VC1 of scientific data are filled in

- move volume checker down with speed level and torque level specified by the command; at the end, the word # 2 VCD and the word # 5 VC2 of scientific data are filled in

- spend the waiting time specified as last but one word of the command

- move volume checker up with speed level C_VCAC_SPEED and torque level C_VCAC_TORQUE; at the end, the word # 2 VCD and the word # 6 VC3 of scientific data are filled in

While executing the <u>VCAC</u> motion and during each motion in direction UP, the "waiting time for completion of each volume checker movement" specified by the command is used as timeout for the motion, and it is used to perform the "Volume checker microswitch check".

The volume checker motion commands require specifying the speed and the torque level to be applied. At page F1 it is reported the mapping between these levels and the RPM mA values (column "speed level used by the speed handler").

The volume checker motor is automatically powered on/off at the start/end of volume checker motion.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

- SD2 mechanical correctness check

During the motion in direction UP, SD2 SW performs the volume checker microswitch check.

Cross reference to telemetry fields

- volume checker displacement: word#2 of scientific data

- volume checker displacement updated by VCAC after first movement (UP direction): word#4 of scientific data

- volume checker displacement updated by VCAC after second movement (DOWN direction): word#5 of scientific data

- volume checker displacement updated by VCAC after third movement (UP direction): word#6 of scientific data

- volume checker microswitch status: word#13 of scientific data

- volume checker motor direction: bit 12 of word#7 of scientific data; bit 12 of word#7 of housekeeping data

- volume checker driver status: bit 8 of word#8 of scientific data; bit 8 of word#8 of housekeeping data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.5 Sampling tube activation

Functional Description

In order to activate the sampling tube mechanism, the command <u>SARE</u> can be used; it commands the release of sampling tube by means of a fixed duration current pulse of 50 [ms].

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

- SD2 mechanical correctness check

Cross reference to telemetry fields

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.6 Emergency Commands

Functional Description

In emergency conditions, the user can use the <u>EMST</u> and the <u>ABRT</u> commands. These commands are not queued by SD2 but are "immediate" commands and are executed as soon as they are received by SD2.

When <u>EMST</u> is received, SD2 executes the following:

- immediate power off of all devices

- stop of all processing

- clear the specific command queue, so that when EMST is received all the queued commands are lost

- communicate the DEAD status into the SD2 backup RAM, enter in DEAD status so that it is necessary to send ENEM command to enter in READY status

When <u>ABRT</u> is received, SD2 executes the following:

- start a deceleration ramp for all the devices in motion status

- power off of all devices

- stop of all processing

- clear the specific command queue, so that when ABRT is received all the queued commands are lost

- communicate the DEAD status into the SD2 backup RAM, enter in DEAD status so that it is necessary to send ENEM command to enter in READY status

When <u>ENEM</u> is received, SD2 executes the following:

- communicate the READY status into the SD2 backup RAM, enter in READY status

The <u>EMST</u>, <u>ABRT</u> and <u>ENEM</u> commands are not nominal commands and can not be included into a mission plan: they require a user intervention.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

Cross reference to telemetry fields

- status flag of all devices: word#7of scientific data; word#7 of housekeeping data

- status of all devices: word#8 of scientific data; word#8 of housekeeping data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.7 WRAD specific command

Functional Description

The <u>WRAD</u> specific command allows a direct access to the SD2 electronic boards registers. When <u>WRAD</u> command is received, SD2 SW performs only the checks relevant to specific command syntax, checksum and SD2 current status. Neither mechanical correctness checks, nor landing gear and tapping station status checks are performed. This means that **the <u>WRAD</u> specific command can be a very dangerous command: only skilled people, only for maintenance reasons shall use it**.

By means of <u>WRAD</u> specific command, it is possible:

- to power on/off a driver (carousel, volume checker, drill translation main winding, drill translation redundant winding, drill rotation, sampling tube)

- to power on/off a resolver (carousel resolver, drill translation resolver)

- to set a speed- torque value

Annex B reports the hexadecimal sequences for powering on/off; the table shows the commands for the SD2 Test Support Software (commands starting by 1002) and commands for SD2 SW (starting by 7ADD). Please note that for driver powering on, two command sequences shall be provided.

Annex C reports the hexadecimal words to be used for setting speed and torque to drill translation. The complete command for setting the speed and torque is

7ADD 1D hex_val cksum (according to Table 2).

The checksum shall be evaluated as sum of 7ADD+1D+hex_val

Annex D reports the hexadecimal words to be used for setting speed and torque to carousel. The complete command for setting the speed and torque is

7ADD 1B hex_val cksum (according to Table 3).

The checksum shall be evaluated as sum of 7ADD+1B+hex_val

Annex E reports the hexadecimal words to be used for setting speed and torque to drill rotation. The complete command for setting the speed and torque is

7ADD 1B hex_val cksum (according to Table 4).

The checksum shall be evaluated as sum of 7ADD+1B+hex_val

Annex F reports the hexadecimal words to be used for setting speed and torque to volume checker. The complete command for setting the speed and torque is

7ADD 1F hex_val cksum (according to Table 5).

The checksum shall be evaluated as sum of 7ADD+1F+hex_val

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

During the motion, SD2 SW does not perform any speed check.

Cross reference to telemetry fields

- status flag of all devices: word#7of scientific data; word#7 of housekeeping data

- status of all devices: word#8 of scientific data; word#8 of housekeeping data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

Examples

1. power on drill translation main winding in direction UP: send to SD2 the following two commands:

7ADD 001C 0001 7AFA

7ADD 001C 0003 7AFC

2. set speed 25,04[rpm] and torque 75[mA] to drill translation : send the command

7ADD 001D 00E0 7BDA

3.2.8 RDAD specific command

Functional Description

The RDAD specific command allows reading the contents of an SD2 electronic boards register. When RDAD command is received, SD2 SW acquires the contents of the specified register, stores the register address in word #9 of scientific data, and stores the register contents in the word # 10 of scientific data.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

Cross reference to telemetry fields

- register address: word#9 of scientific data

- register contents: word#10 of scientific data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.9 General Driver commands

Functional Description

The ZERO specific command moves up the volume checker and moves the carousel and the drill translation to their zero positions. The only parameter of the command is the timeout to be applied to the volume checker motion; in fact, it makes use of the hard coded values of speed and torque levels C_ZERO_DT_SPEED, C_ZERO_DT_TORQUE, C_ZERO_CAR_SPEED, C_ZERO_CAR_TORQUE.

The ONOF specific command switches on/off the devices specified by the command. The status of the device will be "switched on" until:

- another ONOF command with parameter "switch off" will be received by SD2

- a motion command is received which autonomously switches off the device

If the specified parameters of <u>ONOF</u> are "switching on" the "sampling tube", then the sampling tube is activated in continuous mode and shall be powered off explicitly by the <u>ONOF</u> command with parameters "switch off" "sampling tube".

The <u>ACRE</u> specific command switches on the specified resolver (but only if not already powered on), waits for C_COMMAND_PERIOD and then switches off the specified resolver (but only if the resolver status was powered-off before the reception of <u>ACRE</u> command). When the resolver is switched on, its value is reported in the words # 0 and/or 1 of scientific data, and in words # 5 and/or 6 of housekeeping data.

Cautions and warnings

Before starting <u>ZERO</u> command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

- tapping station status check

- SD2 mechanical correctness check

Before starting <u>ONOF</u> and <u>ACRE</u> command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

Cross reference to telemetry fields

- resolver values: words # 0 and 1 of scientific data; words # 5 and 6 of housekeeping data

- status flag of all devices: word#7of scientific data; word#7 of housekeeping data

- status of all devices: word#8 of scientific data; word#8 of housekeeping data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

Examples

1. The command ONOF 0 0 0 1 0 0 0 0 (0x0910 0x0910) switches on the drill translation redundant winding; to switch off it, use ONOF 0 0 0 0 0 0 0 0 0 (0x900 0x900)

3.2.10 LANDG Specific Command

Functional Description

Before starting a drill translation motion, SD2 SW checks if there is no possible collision between the drill and the landing gear legs as detailed in the chapter "Landing Gear Position Check".

The parameters of LANDG specific command are:

- six values A B C D E F defining four forbidden ranges [A,B] [C,D] [E,F] [G,H]. The only constraint for these values is A≤B C≤D E≤F G≤H; the ranges can be overlapped. It is necessary to specify 4 ranges because three are the landing gear legs and one range can be used for handling the passing through the zero position

- the "word index" of the word of the LANDG housekeeping data containing the current position of the lander. The "word index" starts from 0.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

The LANDG command shall be sent to SD2 before starting a drill translation motion since, by default, the four ranges are fixed to [0,0xFFFF], so that any drill translation motion is forbidden.

The six parameters A B C D E F are handled as unsigned words.

The user shall take into account that the units of the parameters A B C D E F and the units of the current landing gear position contained in the LANDG housekeeping data shall be the same.

Cross reference to telemetry fields

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.11 STARTOP STOPOP Specific Commands

Functional Description

As specified in the chapter "SD2 States", SD2 status can be one out of the eight specified. Every time the SD2 status changes, besides the scientific and housekeeping data also the SD2 backup RAM is updated, so that all the other Rosetta instruments can access to the SD2 status.

From the operational point of view there are only three states:

- "undefined", the SD2 SW is not ready to start the processing

- "ready", the SD2 SW executes all its tasks

- "dead", an emergency situation was raised

The other states "drill in progress" .. "carousel motion completed" are, from operational point of view, substatus of "ready". They can be set only by using the specific commands <u>STARTOP</u> <u>STOPOP</u>, as specified in the hereunder table.

Command	SD2 resulting status
STARTOP drilling	drill in progress (0x0F00)
STARTOP sampling	sampling in progress (0x00F0)
STARTOP carousel rotation	carousel in progress (0x000F)
STARTOP SD2 operation	ready (0xF000)
STOPOP drilling	drill completed (0x0A00)
STOPOP sampling	sampling completed (0x00A0)
STOPOP carousel rotation	carousel rotation completed (0x000A)
STOPOP SD2 operation	undefined (0x0000)

Both <u>STARTOP</u> and <u>STOPOP</u> commands require as parameter the "kind of operation". Moreover, the <u>STOPOP</u> command requires the "notify" parameter: if it is set to 1, then at the end of command processing SD2 SW will send the OCPL Request Code to CDMS.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

The <u>STOPOP</u> command is rejected if some device status is power on.

Cross reference to telemetry fields

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.12 LDMP Specific Command

Functional Description

A sequence of SD2 specific commands can be stored into CDMS memory buffer; the <u>LDMP</u> command allows to start the loading of the specific command sequence from CDMS memory buffer into SD2 internal buffer and, at the end, to execute it.

When the <u>LDMP</u> command is received, SD2 starts the loading of the telecommand buffer section stored into CDMS memory buffer: blocks of 32 words are loaded from CDMS memory into SD2 internal buffer until the whole command sequence (whose length is specified in the command) has been loaded; the last block will be the remainder of command sequence length split into blocks of 32 words.

After the loading, SD2 executes the "Adler32 checksum check".

When the loading is successfully completed, SD2 starts the processing of the specific commands contained in the mission plan, in FIFO order.

During command sequence loading, any specific commands different from <u>ABRT</u> and <u>EMST</u> are rejected. In the command sequence to be loaded by means of the <u>LDMP</u> command can not be included the <u>ABRT</u>, <u>EMST</u>, <u>ENEM</u>, <u>LDMP</u> commands.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

- Adler32 checksum check

Cross reference to telemetry fields

- mission plan loading status: word#18 of scientific data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.13 EHEN Specific Command

Functional Description

The <u>EHEN</u> specific command allows to disable/ enable the recovery actions related to a failure detected by SD2.

The parameters of the command are the kind of recovery action to be enabled/ disabled. By default, all the recovery actions are enabled.

See also chapter "SD2 Error Handling" for further details related to errors detected by SD2 and recovery actions.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

Cross reference to telemetry fields

- status of error handling procedure: word#16 of scientific data

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.14 MHIT Specific Command

Functional Description

The <u>MHIT</u> specific command allows the management of the SD2 HITB (Historical Internal Tracking Buffer) that are handled by CDMS as scientific data.

SD2 allocates a buffer of C_SC_DATE_NUM_FRAMES number of scientific data frame, each frame consisting of 32 16-bit words. This buffer is handled as circular buffer. Cyclically, SD2 acquires the telemetry data and fills in this scientific data buffer. By default the acquisition period is C_HITB_DEF_ACQ. The acquisition of scientific data starts autonomously at completion of SD2 SW initialization.

It is possible to change the acquisition period by sending the specific command <u>MHIT</u> with parameters:

- S/H bit set to 1

- ACQ.Period equal to the new value

The dump of the HITB starts only when required, by sending to SD2 the <u>MHIT</u> command with the following parameters:

- S/H bit set to 0

- ACQ.Period equal to any value; also 0 value is accepted, since this data is not taken into account when S/H bit holds 0

When an HITB dump is commanded, SD2 stops the scientific data acquisition and starts the dump. The dump continues until all the acquired data have been transmitted. The dump can not be interrupted, also by

means of emergency commands. When the dump is completed, SD2 re-starts autonomously the acquisition of scientific data by using the last specified acquisition period.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

Cross reference to telemetry fields

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.2.15 DELAY Specific Command

Functional Description

The <u>DELAY</u> specific command allows waiting the time specified by the command before starting the processing and the execution of next specific command.

Cautions and warnings

Before starting command execution, SD2 SW performs the following checks:

- specific command syntax and checksum checks

- command against SD2 status check

Cross reference to telemetry fields

- replica of specific command: words #19..28 of scientific data; word #4 of housekeeping data

- command status: word#15 of scientific data; word#11 of housekeeping data

- SD2 current status: word#17 of scientific data; word#13 of housekeeping data

- error flag: word#16 of scientific data; word#14 of housekeeping data

3.3 SD2 CHECKS

SD2 performs the following checks:

- specific commands syntax checks

- specific command checksum check

- specific command against SD2 status check

- landing gear position check

- tapping station status check

- SD2 mechanical correctness checks

- drill rotation speed check

- drill translation speed check

- carousel speed check

- volume checker microswitch check

- drill translation windings check

- CDMS-SD2 communication deadlock check

- CDMS standard command check

- internal failure check

- Adler32 checksum check, applied only to the <u>LDMP</u> specific command

When one check fails, SD2 executes the recovery actions related to the error severity, as described in the chapter "SD2 Recovery Actions". The failure is communicated to CDMS both in the housekeeping data (word #14) and in the scientific data (word #16). The code of each failure is specified in [CDMSICD] and is listed in the Appendix A.

3.3.1 Specific Commands Syntax Checks

Functional Description

For each specific command, SD2 SW checks that:

- the number of words

- the fixed bits of the command words

- the range of each parameter

- range of parameters

are according to [CDMSICD].

Cautions and Warnings

When SD2 SW detects a syntax check failure, the failures

EC_SPC_CMD_SYNTAX_FAILURE,
EC_SPC_CMD_DATA_OUT_OF_RANGE,
EC_SPC_CMD_ID_FAILURE,
EC_SPC_CMD_WRDC_FAILURE,

with severity 6 is raised. It has been coded this severity level since it has been supposed that all the specific commands which will be sent to SD2 will be prepared in automatic way, so that all the generated commands are correct. The only reason why the received command is not correct is a tele-communication problem

Trouble-shooting

Send the MHIT 0 0 command in order to command the SD2 scientific data dumping. Check if the command echo contained in the words # 19..28 are the expected ones: if so, then there is a problem in the off-line command generation. If the actual and the expected commands are not the same, then there was a problem in the communication.

3.3.2 Specific Commands Checksum Checks

Functional Description

For each specific command, SD2 SW evaluates the checksum according to the following rule

$$Checksum = \sum_{n=1}^{m-1} SSCMD(n)$$

m is the number of words defining the specific command, and compares the evaluated checksum with the word number m of the command.

Cautions and Warnings

When SD2 SW detects a checksum check failure, the failure EC_SPC_CMD_CHECKSUM_FAILURE with severity 6 is raised. It has been coded this severity level since it has been supposed that all the specific commands which will be sent to SD2 will be prepared in automatic way, so that all the generated commands are correct. The only reason why the received command is not correct is a tele-communication problem.

Trouble-shooting

Send the MHIT 0 0 command in order to command the SD2 scientific data dumping. Check if the command echo contained in the words # 19..28 are the expected ones: if so, then there is a problem in the off-line command generation. If the actual and the expected commands are not the same, then there was a problem in the communication.

3.3.3 Specific Commands against SD2 Status Check

Functional Description

When a specific command is received, SD2 SW checks if it is allowed in SD2 current status. The checks are the following:

- when in "dead" status, SD2 SW rejects any specific command with exception of <u>ENEM</u>, <u>MHIT</u>, <u>STOPOP</u> with parameter "SD2 operation"

- when in "undefined" status, SD2 SW rejects any specific command with exception of <u>STARTOP</u> with parameter "SD2 operation"

- while a loading a mission plan, any command different from ABRT and EMST is rejected

Cautions and Warnings

When SD2 SW detects such a check failure, the failure

EC_SPC_CMD_REJECTED_IN_DEAD

EC_SPC_CMD_REJECTED_IN_UNDEFINED

EC_SPC_CMD_REJ_DURING_LDMP

 with severity 1 is raised (only warning message).

Trouble-shooting

Send the MHIT 0 0 command in order to command the SD2 scientific data dumping and check the SD2 status.

3.3.4 Landing Gear Position Check

Functional Description

When a drill translation is commanded, before starting the motion, SD2 performs the LANDING GEAR position check: drill translation is allowed only if the landing gear position is outside the LANDING LEGS forbidden ranges.

In order to know the LANDING GEAR forbidden ranges and the LANDING GEAR position, two steps are required:

- the <u>LANDG</u> specific command shall be sent to SD2: it defines the forbidden ranges and the index I (I starts from 0 and is defined with respect to the 0 offset), in the LANDING GEAR backup RAM where it is contained the actual position of LANDING GEAR

- the LANDING GEAR backup RAM shall be updated by LANDG: it contains (in position I with respect to the 0 offset) the actual LANDING GEAR position

The landing gear position check is performed only if the drill translation is commanded in the range [C_LANDG_MIN, C_LANDG_MAX] (as per Table in the paragraph "SD2 SW constants").

By default, if no <u>LANDG</u> command has been received by SD2, the whole range [0,0xFFFF] is handled as forbidden range; this guarantees that by default it is not allowed any drill translation movement.

Cautions and Warnings

When SD2 SW detects such a check failure, the failure EC_LANDG_POS_CHECK_FAILURE with severity 3 is raised.

<u>The units of LANDING GEAR position in the backup RAM and the forbidden ranges in LANDG command SHALL be the same.</u>

Trouble-shooting

Wait for the correct LANDG positioning and then re-send the command, or disable the recovery action by issuing the command EHEN and re-send the command.

Examples

command sequence 1:

150 1 2 3 4 5 6 7 8 9 – contents of landing gear backup RAM

.....

DRTR 31 2 50000 -- drill translation towards 500 [mm]

...

The DRTR command is rejected because, by default, all the LANDING GEAR positions are rejected: SD2 enters in emergency status

command sequence 2:

150 1 2 3 4 5 6 7 8 9 – contents of landing gear backup RAM.

....

LANDG 0 1 100 200 300 1000 2000 10000 20000   -- 0 is the index in the landing gear backup RAM containing the landing gar position; [1,100] [200, 300] [1000, 2000] [10000, 20000] are the forbidden ranges

DRTR 31 2 50000 -- drill translation towards 500 [mm]

...

The DRTR is accepted because the current landing gear position (150) is outside the forbidden ranges

command sequence 3:

250 1 2 3 4 5 6 7 8 9 – contents of landing gear backup RAM.

....

LANDG 0 1 100 200 300 1000 2000 10000 20000   -- 0 is the index in the landing gear backup RAM containing the landing gar position; [1,100] [200, 300] [1000, 2000] [10000, 20000] are the forbidden ranges

EHEN 0 1 1   -- the recovery action related to error with severity 3 is disabled, the recovery actions related to error with severity 5 and to drill translation speed check failure are enabled

DRTR 31 2 50000 -- drill translation towards 500 [mm]

...

The DRTR is accepted because the current landing gear position (250) is inside the forbidden ranges and this causes the landing gear position check failure, but the EHEN command disables the relevant recovery action.

command sequence 4:

250 1 2 3 4 5 6 7 8 9 – contents of landing gear backup RAM.

....

DRTR 31 2 500 -- drill translation towards 5 [mm]

....

The drill translation is accepted despite of the landing gear position (by default, all the positions are forbidden) because the motion is over the balcony.

3.3.5 Tapping Station Status Check

Functional Description

When a carousel motion is commanded, before starting the motion, SD2 SW checks if both the COSAC and PTOLEMY tapping stations are disengaged.

The status of the tapping stations is known by access to the COSAC and PTOLEMY backup RAMs.

The following status are handled by SD2 SW:

- COSAC disengaged: value of COSAC backup RAM 0xFFFF

- COSAC engaged: value of COSAC backup RAM 0xAAAA

- COSAC undefined: any other value of COSAC backup RAM

- PTOLEMY disengaged: value of PTOLEMY backup RAM 0xAAAA

- PTOLEMY engaged: value of PTOLEMY backup RAM 0xFFFF

- PTOLEMY undefined: any other value of PTOLEMY backup RAM

Cautions and Warnings

When SD2 SW detects such a check failure, the failure EC_COSAC_STATUS_ENGAGED, EC_COSAC_STATUS_UNDEFINED,   EC_PTOLEMY_STATUS_ENGAGED, EC_PTOLEMY_STATUS_UNDEFINED with severity 3 is raised.

Trouble-shooting

Wait for the correct tapping status and then re-send the command, or disable the recovery action by issuing the command EHEN and re-send the command.

Examples

command sequence 1:

FFFF – contents of COSAC backup RAM: disengaged

AAAA – contents of PTOLEMY backup RAM: disengaged

...

CAPO 31 6 11400 20 – move carousel in the position 11400 arcmin

...

The CAPO command is accepted because both the tapping stations are disengaged.

command sequence 2:

AAAA – contents of COSAC backup RAM: engaged

AAAA – contents of PTOLEMY backup RAM: disengaged

...

CAPO 31 6 11400 20 – move carousel in the position 11400 arcmin

...

The CAPO command is rejected because COSAC is engaged and PTOLEMY is disengaged.

command sequence 3:

AAAA – contents of COSAC backup RAM: engaged

AAAA – contents of PTOLEMY backup RAM: disengaged

...

EHEN 0 1 1   -- the recovery action related to error with severity 3 is disabled, the recovery actions related to error with severity 5 and to drill translation speed check failure are enabled

CAPO 31 6 11400 20 – move carousel in the position 11400 arcmin

...

The CAPO command is accepted because COSAC is engaged and PTOLEMY is disengaged, and this causes the check fails, but the EHEN command disables the relevant recovery action.

3.3.6 SD2 Mechanical Correctness Checks

Functional Description

Before starting a motion, SD2 SW checks if there is not interference with the SD2 mechanical unit. These are the performed checks:

- check relevant to <u>drill translation</u> motion: before drill translation motion, SD2 checks the commanded drill position with respect to the actual carousel position and allows the motion only in the following cases:

    - if the carousel is in zero, the command is accepted whatever the commanded position

    - if one out the 26 ovens is under the drill axis, the command is accepted if the drill translation commanded position is over the bottom of the oven

    - if the re-arm oven is under the drill axis, the command is accepted if the drill translation commanded position is over the bottom of the re-arm oven

    - if the carousel is in any other position, the command is accepted if the drill translation commanded position is over the top of the ovens

- check relevant to <u>carousel</u> rotation: carousel position command is accepted only if the drill translation is in zero position

- check relevant to <u>volume checker</u> motion: volume checker motion in direction up is always accepted. Volume checker motion with direction down is possible only if one out the 26 ovens is under the volume checker axis, or when the zero carousel position is under the volume checker axis

- check relevant to <u>drill rotation</u> motion: drill rotation command is accepted only if the zero carousel position is under drill axis

- check relevant to <u>sampling tube</u> release: the SARE command is accepted only if either the zero carousel position is under drill axis or the drill is in zero position

Each check is performed with tolerance. Tolerance values and exact ranges are specified in the paragraph "SD2 SW constants".

Cautions and Warnings

When SD2 SW detects such a check failure, the following failures are raised:

EC_SARE_CONFIGURATION_CHECK_FAILURE,

EC_DRILL_POS_CHECK_FAILURE

EC_CAROUSEL_POS_CHECK_FAILURE

All these failures have severity 4: they can not be disabled.

The checks are not performed when the command <u>ZERO</u> is executed.

Trouble-shooting

Send the MHIT 0 0 command in order to command the SD2 scientific data dumping and analysis the SD2 status. These failures can not be disabled, so the only way to move a SD2 mechanical unit is to use the "direct commands" (see chapter 2).

3.3.7 Drill Rotation Speed Check

Functional Description

When the drill rotation command is under execution and the drill has completed the acceleration ramp, SD2 SW evaluates the drill rotation speed by using the drill rotation counter, writes it in the word #3 of Scientific Data, and checks if the evaluated speed is equal to the commanded one.

The check is performed only if the commanded speed is higher than a fixed value. This value and the tolerance value are reported in the paragraph "SD2 SW constants".

Cautions and Warnings

When SD2 SW detects such a failure, the failure EC_CHK_DRILLROT_SPEED_FAILURE is raised, with severity 3: it can be disabled.

Trouble-shooting

Check the drill rotation mechanical-electrical unit. Send the EHEN command in order to disable the recovery action related to the check failure.

3.3.8 Drill Translation Speed Check

Functional Description

When the drill translation command is under execution and the drill has completed the acceleration ramp (see [URD] for further details on the drill translation control), SD2 SW evaluates the drill translation speed by using sampled resolver positions and checks if the evaluated speed is equal to the commanded one.

The check is performed only if the commanded speed is higher than a fixed value. This value and the tolerance value are reported in the paragraph "SD2 SW constants".

Cautions and Warnings

When SD2 SW detects such a failure, the failure EC_CHK_DRILLTRA_SPEED_FAILURE is raised, with severity 2.

Trouble-shooting

Check the drill rotation mechanical-electrical unit. Send the EHEN command in order to disable the recovery action related to the check failure.

3.3.9 Carousel Speed Check

Functional Description

When the carousel motion command is under execution and the carousel has completed the acceleration ramp (see [URD] for further details on the carousel speed control), SD2 SW evaluates the carousel speed by using sampled resolver positions and checks if the evaluated speed is equal to the commanded one.

The check is performed only if the commanded speed is higher than a fixed value. This value and the tolerance value are reported in the paragraph "SD2 SW constants".

Cautions and Warnings

When SD2 SW detects such a failure, the failure EC_CHK_CAROUSEL_SPEED_FAILURE is raised, with severity 3: it can be disabled.

Trouble-shooting

Check the carousel mechanical-electrical unit. Send the EHEN command in order to disable the recovery action related to the check failure.

3.3.10 Volume Checker Microswitch Check

Functional Description

When the volume checker is commanded to move in direction "up", it is checked that the upper microswitch of the volume checker is reached within the timeout specified in the MVCK / VCAC commands. If the timeout is expired, the motion continues again another time: if the upper microswitch of the volume checker is not reached again within the timeout specified in the command, the failure EC_VCK_MICROSW_CHECK_FAILURE is raised.

Cautions and Warnings

The failure EC_VCK_MICROSW_CHECK_FAILURE has severity 3, and can be disabled by means of EHEN command.

Trouble-shooting

Send the MVCK/ VCAC command with a higher torque value. Send the EHEN command in order to disable the recovery action.

3.3.11 Drill Translation Windings Check

Functional Description

SD2 allows checking the drill translation windings before starting a mission requiring the drill translation motion. This check can be useful at Rosetta system level when, after landing on the comet, it will be necessary to schedule the sequence of missions to be performed by all the Rosetta sub-systems: if this check points out that both the windings are correctly working, the SD2 mission can start whenever you want. If in turn this check points out failures in the windings, it could happen that during the true SD2 mission the two windings do not work at all, perhaps when the drill has overcome the balcony: in this condition it could be dangerous or impossible to execute the missions of the other Rosetta sub-systems.

SD2 allows checking the drill translation windings by providing the specific command DRTC. The DRTC performs the drill translation movement by powering either the main or the redundant winding, according to the input parameter. The movement is towards the position specified in the specific command but with timeout: if the time elapses before reaching the specified position, SD2 declares failed the commanded movement. The failure EC_DRTR_MAIN_MOTION_CHECK_FAILURE is raised when the main winding is powered on and the commanded position is not reached within the specified timeout. The failure EC_DRTR_RED_MOTION_CHECK_FAILURE is raised when the redundant winding is powered on and the commanded position is not reached within the specified timeout.

During the execution of DRTC specific command, the check specified in the chapter "Drill Translation Speed Check" is not performed.

In order to check both the windings, it is suggested to issue at first the DRTC command towards 4 [1/100 mm] (over the oven top) by commanding the main winding, and then to issue again the DRTC command towards zero position by commanding the redundant winding.

Cautions and Warnings

The failures

EC_DRTR_MAIN_MOTION_CHECK_FAILURE and

EC_DRTR_RED_MOTION_CHECK_FAILURE

are raised when the motion performed by powering main/ redundant winding respectively is not completed within the timeout specified in the DRTC specific command.

Trouble-shooting

- Check if the timeout specified in the DRTC command really allows to complete the specified motion

- Send the DRTC command with a higher torque value

Send the ENEM command in order to exit from SD2 DEAD status.

3.3.12 CDMS-SD2 communication deadlock check

Functional Description

Every time SD2 needs to communicate with CDMS, according to a master-slave protocol SD2 resets the SR (Service Request Flag) and waits for the TRQC (Transmit Request Code) action from CDMS. In order to avoid a deadlock, SD2 is waiting for the TRQC with a timeout. If the timeout elapses, SD2 notifies the failure EC_XXXX_TIMEOUT_EXPIRED (XXXX stands for the name of the request code) and automatically starts a retry-mechanism: it resets the SR flag and waits, with timeout, for the TRQC. This mechanism is applied for, as a maximum, C_CDMS_NUM_RETRY times.

If for C_CDMS_NUM_RETRY times the EC_XXXX_TIMEOUT_EXPIRED is raised again, the failure EC_XXXX_MAX_NUM_RETRY_REACHED (XXXX stands for the name of the request code) is raised and the request is not anymore raised.

Values of C_CDMS_NUM_RETRY and timeout (C_CDMS_TIMEOUT) are reported in chapter "SD2 SW Constants".

Cautions and Warnings

When SD2 SW detects such a failure, the following failures are raised (according to the failed request):

EC_SRDY_TIMEOUT_EXPIRED

EC_FLSP_TIMEOUT_EXPIRED

EC_OCPL_TIMEOUT_EXPIRED

EC_SCMD_TIMEOUT_EXPIRED

EC_RDBF_TIMEOUT_EXPIRED

EC_WRBF_TIMEOUT_EXPIRED

All of them have severity 1, only warning message.

When the failure continues for , the following failures are raised:

EC_SRDY_MAX_NUM_RETRY_REACHED

EC_FLSP_MAX_NUM_RETRY_REACHED

EC_OCPL_MAX_NUM_RETRY_REACHED

EC_SCMD_MAX_NUM_RETRY_REACHED

EC_RDBF_MAX_NUM_RETRY_REACHED

EC_WRBF_MAX_NUM_RETRY_REACHED

Trouble-shooting

Check the CDMS-SD2 communication.

3.3.13 CDMS standard command check

Functional Description

The communication between SD2 SW and CDMS is based on exchanging of words containing action codes, request codes, data (see [CDMSICD]) and only a subset of them are used by SD2 SW. When SD2 SW receives an action code different from the foreseen ones, a failure is raised.

Cautions and Warnings

The following failures can be raised:

EC_RERC_DESTINATION_UNIT_OFF

EC_RERC_SCIENCE_DATA_OUT

EC_RERC_MEMORY_FULL

EC_RERC_REQ_UNDUE

EC_RERC_ILLEGAL_ADDRESS

EC_RERC_ILLEGAL_REQ_CODE

EC_RERC_REASON_UNKNOWN

EC_RASV_RECEIVED

EC_RAXT_RECEIVED

EC_RSCS_RECEIVED

EC_RBUS_RECEIVED

EC_TRG_RECEIVED

EC_USR_CMD_ILLEGAL_AC

EC_USR_DATA_ILLEGAL_AC

EC_STBY_RECEIVED

EC_RCMS_RECEIVED_WITH_NO_MP_LOAD

EC_RMOD_NORMAL_MODE

EC_RMOD_NO_NORMAL_MODE

EC_RSST_INIT_RECOVERY_PROC

EC_RSST_NO_RECOVERY_PROC

Trouble-shooting

Check the CDMS- SD2 SW communication.

3.3.14 Internal Failure Check

Functional Description

SD2 SW performs checks on the internal data flow (as an example, at the end of each step it checks that the stack is always empty; when working on arrays, it checks that the index is always within the array range), on mechanical device status

Cautions and Warnings

When such a failure is detected, the errors EC_XXX_INTERNAL_ERROR (where XXX is an abbreviation of source module name), EC_Q_YYY_FULL_ERROR (where YYY is an abbreviation of queue source module name), EC_BH_SWITCHONOFF_ERROR (error during mechanical device switch ON/OFF), EC_BH_SETSTD_ERROR (error in the set bit of electronics), EC_BH_SWITCHTUBE_ERROR (error in the handling of sampling tube), EC_SH_HW_ERROR (error detected by speed-handler source module while handling the electronics) are raised. All these failures have severity 6: they can not be disabled

Trouble-shooting

All these failures point out that something is wrong or in the software (software bugs) or in the hardware. Send the MHIT command in order to dump the acquired scientific data and analyze the data.

3.3.15 Adler32 Checksum Check

Functional Description

When the <u>LDMP</u> specific command is received by SD2 and the loading is completed, SD2 evaluates the checksum of the loaded commands according to Adler32 algorithm and compares it with the expected Adler32 checksum provided by the <u>LDMP</u> command. If check fails, an error flag is reported in telemetry, the SD2 internal buffer containing the command sequence is cleared and SD2 re-starts the whole command sequence loading again. The re-loading can be repeated up to C_LOAD_MAX_NUM times.

The Adler32 of the mission plan can be evaluated as suggested in [SDE]- chapter "Test Conduction", paragraph "Creation of a mission plan".

Cautions and Warnings

When such a failure is detected, the error EC_MP_ADLER32_CKSUM_FAILURE is raised and SD2 re-starts the loading up to three times. After three failures, the error EC_MP_LOAD_FAILED is raised, the loading is stopped and the buffer containing the mission plan is cleared.

Trouble-shooting

Check the Adler32 of the LDMP command or check the CDMS- SD2 SW communication.

3.4 SD2 ERROR HANDLING

3.4.1 SD2 Detected Errors

The failures detected by SD2 SW are reported in telemetry both as housekeeping data (word # 14) and as scientific data (word # 16). To each failure, a severity error level is associated and for each severity SD2 SW performs a recovery action.

The nibbles of the hexadecimal error code have the following meaning:

- thirst three nibbles: error identifier

- less significant nibble: error severity.

The recovery actions related to the detected errors are specified in the chapter "SD2 Recovery Actions" while the Appendix A reports the complete list of detected errors.

3.4.2 SD2 Recovery Actions

SD2 autonomously executes the recovery action related to the detected failure. Only four recovery actions are implemented:

- nothing to be done: the SD2 SW continues its processing as no failure raised

- autonomous execution of the actions related to ABRT command. As a consequence SD2 enters in DEAD status

- autonomous execution of the actions related to EMST command. As a consequence SD2 enters in DEAD status

- recovery action applied only when drill translation speed check failure is detected: as first trial the drill translation is commanded with a fixed high torque; if check fails again, then both the main and the redundant windings are powered on and the fixed torque is used; as last trial, the actions related to ABRT command are executed.

The recovery action executed by SD2 SW is related to the severity of the detected error.

For some error it is possible to disable the recovery action. For this reason, the severity is marked either as "selectable" (the relevant recovery action can be disabled), or as "always selected" (the relevant recovery action can not be disabled)

The failure severities are the following

> 0=ignore; always selected

> 1=warning; always selected

> 2=redundancy_check; selectable

> 3=soft; selectable

> 4=soft; always selected

> 5=hard; selectable

> 6=hard; always selected

Severity 0. When the severity is "ignore", the error is not reported in telemetry data and no recovery action is executed: the error condition is ignored and SD2 SW continues its processing.

Severity 1. When a "warning" error condition is detected, the error is reported in telemetry and no other action is performed.

Severity 3 and 4. When a "soft" emergency error is detected, either "selectable" or "always selected", the relevant error code is reported in telemetry and following recovery procedure is executed:

- the actions relevant to ABRT command are executed. As a consequence, SD2 enters in Dead status.

Severity 5 and 6. When a "hard" emergency error is detected, either "selectable" or "always selected", the relevant error code is reported in telemetry and the following recovery procedure is executed:

- the actions relevant to EMST command are executed. As a consequence, SD2 enters in Dead status.

Severity 2. The error with severity 2 is relevant to drill translation speed check. The relevant error is reported in telemetry and the following recovery action is executed:

- first trial: the drill motion continues by using the fixed recovery torque C_DT_RECOVERY_TORQUE instead of the torque specified by the command.

- second trial: if the check fails again, the drill translation continues by using the fixed recovery torque value and by powering both main and redundant drivers.

- last trial: if the check fails again, the actions relevant to ABRT command are executed and, as a consequence, SD2 enters in Dead status.

The recovery action applied when drill translation speed check is detected is reported in the word #7 STFG of Scientific data, bits SF9 SF10 SF11.

Examples

1. The failure EC_SPC_CMD_DATA_OUT_OF_RANGE has code 0x0016: severity 6, the actions relevant to EMST command are executed, SD2 SW enters in "dead" status. The recovery action can not be disabled.

2. The failure EC_VOLUME_COUNTER_INVALID has code 0x06A3: severity 3, the actions relevant to ABRT command are executed, SD2 SW enters in "dead" status. The recovery action can be disabled

3.4.3 How to disable the SD2 Recovery Actions

As already stated in chapter "SD2 Recovery Actions", the recovery actions marked as "selectable" can be disabled. For doing that, it shall be sent to SD2 the specific command EHEN with the following parameters (see [CDMSICD]):

- RF bit shall be set to 0 in order to disable the recovery action related to the error severity 2; it shall be set to 1 in order to enable it

- Hf bit shall be set to 0 in order to disable the recovery action related to error severity 5; it shall be set to 1 in order to enable it

- Sf bit shall be set to 0 in order to disable the recovery action related to error severity 3; it shall be set to 1 in order to enable it

Disabling a recovery action does not mean to disable the relevant check. In fact, when a recovery action is disabled, the SD2 SW continues to execute all the checks, reports the failure in telemetry if the severity of the detected error is not 0, but does not execute the actions related to the disabled recovery action and continues its processing as no failure raised.

By default, all the recovery actions are enabled.

The actual status of recovery actions is reported in word #14 EHSTATUS of scientific data. By default EHSTATUS value is 7: all recovery actions enabled.

It is worth to notice that the EHEN disables/ enables the recovery action related to ALL the errors with the same severity, i.e. it is not related to just one specific error. It is so suggested to send EHEN twice, once before sending the specific command causing the failure in order to disable the recovery action, and once after the specific command execution in order to enable the recovery action.

The severity of an error is hard coded: the only way to change it is to modify the SD2 source code.

Examples

1. The EHEN with Sf bit set to 0 disables the recovery action related to the errors:

    - 0403: drill rotation speed failure

    - 0423: carousel speed failure

    - 0613: COSAC tapping station is engaged

    - 06B3: LANDG position failure

    and so on.

    To send EHEN with Sf bit set to 0 means that if drill rotation speed failure is detected, the error is reported in telemetry but the drill rotation is not stopped, it continues

2. If you want to move the carousel despite of the contents of COSAC/ PTOLEMY backup RAM, you have to send the EHEN with bit Sf set to 0, but you have to pay attention to do it: you should send it just before to command the carousel motion and to send again just after the completion of carousel motion for enabling the recovery action. In fact EHEN with Sf bit set to 0 disables also the recovery action related to, as an example, the landing gear position failure

3.5 SD2 SW CONSTANTS

In the following, the table contains the values of the SD2 SW constants are listed. These constants are used by SD2 SW for executing the above listed checks.

Name	Meaning	Value	Note
C_ZECTO	zero position carousel rotation tolerance	### 2.5 deg	
C_DOVTO	drill axis to oven axis position tolerance	### 20 arcmin	
C_DDUTO	drill axis to dummy oven axis position tolerance	### 10 arcmin	
C_ZEDTO	zero position drill translation tolerance	> 0.5 mm	
C_MRFR	Motion regulation frequency	4 Hz	
C_CDMS_TIMEOUT	Timeout for CDMS successful communication	30 [250 ms]	
C_CDMS_NUM_RETRY	maximum number of retries to be used in the CDMS communication protocol	3	
C_COMMAND_PERIOD	period of specific command processor	250 [ms]	
C_LOAD_MAX_NUM	number maximum of mission plan re-load	3	
C_HITB_DEF_ACQ	Default HITB acquisition period	4 [s]	
C_SC_DATE_NUM_FRAMES	maximum number of scientific data frames contained into HITB buffer	256	
C_ZERO_DT_SPEED	Speed of drill translation for executing the ZERO command	speed level 10: 112RPM	
C_ZERO_DT_TORQUE	Torque of drill translation for executing the ZERO command	torque level 2: 225mA	
C_ZERO_CAR_SPEED	Speed of carousel for executing the ZERO command	speed level 5: 50 RPM	
C_ZERO_CAR_TORQUE	Torque of drill translation for executing the ZERO command	torque level 4: 375 RPM	
C_MAX_CDMS_MESSAGE	Dimension of specific command queue	10 (number of specific commands)	
C_MAX_MP_WORDS	Maximum dimension of stored telecommand sequence	100 [16-bit word]	
C_RESOLVER_POWER_DELAY	Time required for resolver power on	50 [ms]	

C_VOLCHK_SWITCH_PERIOD	acquisition period of upper volume checker micro-switch	50 [ms]	
C_VCAC_TORQUE	torque to be used in VCAC command for volume checker up motion	torque level 7: 600 mA	
C_VCAC_SPEED	speed to be used in VCAC command for volume checker up motion	speed level 11: 406RPM	
C_SRDT	Drill translation sampling range, as specified in UR-3.1.2.2.1-1	[0,62500]	1/100 mm
C_DRTD	Drill translation discharge range, as specified in UR-3.1.2.2.1-1	[0,1200]	1/100 mm
C_ARDT	Drill translation re-arm range, as specified in UR-3.1.2.2.1-1	[0,1700]	1/100 mm
C_DFDT	Drill translation default range, as specified in UR-3.1.2.2.1-1	[0,100]	1/100 mm
C_LANDG_MIN, C_LANDG_MAX	Range causing interference with landing gear legs	[0,62500]	1/100 mm
C_CAR_RAMP_DOWN_TH	Threshold used for carousel position control. When $|Car\_Cur\_Pos – Car\_Target\_Pos|$ <= C_CAR_RAMP_DOWN_TH, then the carousel decelerating phase starts	300	arcmin
C_CAR_LOWER_SPEED	Carousel lower speed level to be used during the carousel deceleration phase	speed level 2: 12,5RPM	
C_CAR_TARGET_TH	Threshold used for checking the carousel end of motion	3	Arcmin
C_CHK_DR_MIN_SPEED	Minimum drill rotation speed used for drill rotation measurement	>speed level 10: >407RPM at motor side	
C_CHK_DR_PERIOD	Execution period of drill rotation check	4	s
C_CHK_DR_SPEED_TOL	Tolerance used for drill rotation speed check	6 RPM, after whole gear transmission	
C_NUM_PULSE_PER_ROUND	number of pulses for round of drill rotation	3	int
C_CHK_DT_MIN_SPEED	Minimum drill translation speed used for drill translation speed check	>level speed 3: >37.5 RPM	
C_CHK_DT_PERIOD	Execution period of drill translation check	15 s	

C_CHK_DT_SPEED_TOL	Tolerance used for drill translation speed check	70%	
C_CHK_CAR_MIN_SPEED	Minimum carousel speed used for carousel speed check	>speed level 2: >25 RPM	
C_CHK_CAR_PERIOD	Execution period of carousel check	2 s	
C_CHK_CAR_SPEED_TOL	Tolerance used for carousel speed check	70%	
C_CAR_RAMP_DOWN_MAX_DURATION	Max duration of carousel ramp down phase	60 s	
C_DT_RECOVERY_TORQUE	Torque to be used when drill translation speed check fails first time	torque level 4: 625mA	
C_SECTION_SWITCH_DELAY	delay between switching of two electronics section	5 ms	
C_SET_SPEED_DELAY	delay after set-speed settings	5 ms	
C_POWER_SWITCH_DELAY	delay after power section switching	5 ms	
C_X0	drill position used in the UR-3.1.2.2.2-1	1000	1/100 mm

3.6 SD2 TIMING HANDLING

SD2 SW maintains an internal timing, which starts up at SD2 power on. This time is used both to synchronize the SD2 SW tasks, and to date the telemetry data, both scientific and the housekeeping frames.

The time reported in telemetry is evaluated by SD2 as the time received by means of the RTIM CDMS standard command updated by the time elapsed between the reception of last RTIM command and the time when the telemetry data started.

The time is reported in word#12 of housekeeping data (units is [s]), and in words #11 and 12 of scientific data (units is 31.25[ms]).

4. OPERATIONAL CONSTRAINTS

In order to utilize correctly SD2, it's mandatory to pay attention to the following instructions.

4.1 FM AND FS

- Not use in air. It is request that all the operations are performed in vacuum, because the lubricant does not bear the humidity.
- Not execute the 'locking SD2 for launching' procedure at T<10°C.
- Execute at Troom the 'locking SD2 for transportation' procedure before every transportation.
- Before carousel rotation make sure that all the Tapping Stations are in 'open position'.
- Release Sampling Tube when there is no movement of SD2 devices in progress: the carousel, the drill translation, the drill rotation and the volume checker are at rest
- Before starting drill translation motion, make sure that there is no interference with landing gear legs

4.2 GRM

- Not execute the 'locking SD2 for launching' procedure at T<10°C.
- Execute at Troom the 'locking SD2 for transportation' procedure before every transportation.
- Before carousel rotation make sure that all the Tapping Stations are in 'open position'.
- Release Sampling Tube when there is no movement of SD2 devices in progress: the carousel, the drill translation, the drill rotation and the volume checker are at rest
- Before starting drill translation motion, make sure that there is no interference with landing gear legs

5. SD2 SUBSYSTEM MAINTENANCE PROCEDURES

5.1 SD2 LOCKING

Before SD2 transportation and before SD2 launching, the SD2 S/S shall be in "locked" position. As far as the drill translation is concerned, the "locked" position is some tenths of mm above the drill translation zero position. Since the drill translation motion commands allow to move the drill in the range [0,625] mm, in order to lock SD2 it is necessary to use the specific command WRAD. Since this command could be very dangerous because it allows to write every thing you want into the SD2 electronics boards**, it is mandatory to execute the locking procedure exactly as hereunder specified**. **Before starting the procedure, MAKE SURE that:**

- **there is no interference between the drill translation motion and the actual carousel position**

- **the drill translation starting position is 0.3±0.1 [mm].** In fact next procedures move the drill translation for a fixed time, sufficient for covering the distance between 0.3 [mm] and the locking position

It is MANDATORY that this procedure SHALL be executed at ambient temperature. In fact, at –150° the drill is reduced by about 2 mm; if the locked position were commanded at this temperature, the whole structure will crash when the ambient temperature will be reached.

The procedures consist of 5 steps. The commands can be sent in sequence without delays between two commands (the necessary delay times are already included in the procedures). **Due to the criticality of the locking procedures, the resolver values acquired in steps 1, 3, 5 below defined should be notified to Tecnospazio for making sure that every thing is OK.**

5.1.1 LOCKING SD2 FOR LAUNCH

1. Check starting position Power on the resolvers. Delay 8 [s] Acquire resolver values Power off resolvers Check resolver values to verify that carousel is in zero position and the drill translation position is 0.3±0.1[mm]	0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
2. Move up drill translation for 50 [s] Power on drill translation motor, up direction Set speed=125rpm and current=225mA Wait for 50 [s] Set speed to 0 power off driver	7ADD 001C 0001 7AFA 7ADD 001C 0003 7AFC 7ADD 001D 00A2 7B9C ABEF 00C8 ACB7 7ADD 001D 00F9 7BF3 7ADD 001C 0000 7AF9
3. Check reached position Power on the resolvers. Delay 8 [s] Acquire resolver values Power off resolvers Verify that the translation position is –0.59mm with tolerance +0.1mm –0.4mm	0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
4. Move up drill translation for 20 [s] (to make sure that locking position has been reached) Power on drill translation motor, up direction Set speed=125rpm and current=225mA Wait for 20 [s]	7ADD 001C 0001 7AFA 7ADD 001C 0003 7AFC 7ADD 001D 00A2 7B9C ABEF 0050 AC3F

Set speed to 0 power off driver	7ADD 001D 00F9 7BF3 7ADD 001C 0000 7AF9
5. Check reached position Power on the resolvers. Delay 8 [s] Acquire resolver values Power off resolver Verify that the translation position is equal to the previous one with tolerance +0mm –0.1mm. If not so, repeat all the commands specified for step 4 (from "Power on drill translation motor, up direction" up to "power off driver"), repeat all the commands specified at step 5 (from "Power on the resolvers" up to "Power off resolver"), and verify that the translation position is equal to the previous one with tolerance +0mm –0.1mm. If not so again, contact Tecnospazio staff	0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900

5.1.2 LOCKING SD2 FOR TRANSPORTATION

1. Check starting position Power on the resolvers. Delay 8 [s] Acquire resolver values Power off resolvers Check resolver values to verify that carousel is in zero position and the drill translation position is 0.3±0.1[mm]	0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
2. Move up drill translation for 50 [s] Power on drill translation motor, up direction Set speed=125rpm and current=150mA Wait for 50 [s] Set speed to 0 power off driver	7ADD 001C 0001 7AFA 7ADD 001C 0003 7AFC 7ADD 001D 00A1 7B9B ABEF 00C8 ACB7 7ADD 001D 00F9 7BF3 7ADD 001C 0000 7AF9
3. Check reached position Power on the resolvers. Delay 8 [s] Acquire resolver values Power off resolvers Verify that the translation position is –0.37mm with tolerance +0.1mm –0.4mm	0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
4. Move up drill translation for 20 [s] (to make sure that locking position has been reached) Power on drill translation motor, up direction Set speed=125rpm and current=150mA Wait for 20 [s] Set speed to 0 power off driver	7ADD 001C 0001 7AFA 7ADD 001C 0003 7AFC 7ADD 001D 00A1 7B9B ABEF 0050 AC3F 7ADD 001D 00F9 7BF3 7ADD 001C 0000 7AF9
5. Check reached position Power on the resolvers. Delay 8 [s] Acquire resolver values Power off resolver Verify that the translation position is equal to the previous one with tolerance +0mm –0.1mm.	0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900

If not so, repeat all the commands specified for step 4 (from "Power on drill translation motor, up direction" up to "power off driver"), repeat all the commands specified at step 5 (from "Power on the resolvers" up to "Power off resolver"), and verify that the translation position is equal to the previous one with tolerance +0mm –0.1mm. If not so again, contact Tecnospazio staff	

5.2 SD2 DE-LOCKING PROCEDURE

Before starting the TV tests and before starting the SD2 missions on the comet, SD2 shall be de-locked. At first, the SD2 Health Check specified in [LFT] shall be executed in order to verify the SD2 zero position. Then, the following procedure shall be executed:

- **LANDG housekeeping data shall be properly initialized**

Set Landing gear data by issuing the LANDG specific command. Up to now, the parameters of this command are not known: **they shall be defined at Rosetta system level before launching.** **As an example**, Index 0 ranges [1 100] [200 300] [1000 2000] [10000 20000]	**To Be Specified at Rosetta system level** B0E0 0001 0064 00C8 012C 03E8 07D0 2710 4E20 3421
Move drill translation down to 0.3 mm, 125 RPM 375 mA	296A 001E 2988
Power on the resolvers and check the drill/ carousel position Power on the resolvers. Delay 8 [s] Verify that the translation position is 0.3mm±0.2mm; if not so, contact Tecnospazio staff	0960 0960 ABEF 0020 AC0F 8800 8800

5.3 SD2 DRILL TRANSLATION RESOLVER ZEROING

Performing the following steps can zero the drill translation resolver:

1. drill translation tip shall be put in a fixed position, TBD mm over the base plane.

2. send the following specific commands:

       7ADD 1E 1804 92FF -- power on resolver

       7ADD 1E 180C 9307   -- start trigger

       7046 1F 7065   -- read raw data

       8800 8800   -- dump the scientific data

3. Wait for the end of dump

4. Send again the command

       8800 8800

5. Read in the last frame of scientific data the word number 10, starting from 0, and turn it in decimal. Let it be the VAL value.

6. Evaluate the rule:

       OFFSET= HEX(-(VAL * 2025/8))

       Split it in two parts each one made of 4 digits: high and low part

7. Turn the high part into decimal: let it be ZERO_HIGH; turn the low part into decimal: let it be ZERO_LOW.

8. Power off SD2 electronics.

9. Modified, by means of the forth editor, the bh_const.scr file.

Change the value of the constant BH_DRT_OFFSET_1H to be equal to ZERO_HIGH; change the value of the constant BH_DRT_OFFSET_1L to be equal to ZERO_LOW.

Save the file and re-build SD2 SW.

Example

Let it be VAL equal to 1223 (decimal).

Then, OFFSET =0xFFFB46BC, ZERO_HIGH=dec(FFFB)=65531, ZERO_LOW=dec(46BC)=18108.

The file bh_const.scr should be updated according to the following:

18108 EQU BH_DRT_OFFSET_1L

65531 EQU BH_DRT_OFFSET_1H

5.4 SD2 CAROUSEL RESOLVER ZEROING

Performing the following steps can zero the carousel resolver:

1. carousel shall be mounted in such a way that the oven whose position is 18 is exactly under the volume checker axis

2. send the following specific commands:

   7ADD 1E 1001 8AFC -- power on resolver

   7ADD 1E 1003 8AFE   -- start trigger

   7046 1F 7065   -- read raw data

   8800 8800   -- dump the scientific data

3. Wait for the end of dump

4. Send again the command

   8800 8800

5. Read in the last frame of scientific data the word number 10, starting from 0, and turn it in decimal. Let it be the VAL value.

6. Evaluate the rule:

   OFFSET= HEX(-(VAL * 675/8))

   Split it in two parts each one made of 4 digits: high and low part

7. Turn the high part into decimal: let it be ZERO_HIGH; turn the low part into decimal: let it be ZERO_LOW.

8. Power off SD2 electronics.

9. Modified, by means of the forth editor, the bh_const.scr file.

Change the value of the constant BH_CAR_OFFSET_1H to be equal to ZERO_HIGH; change the value of the constant BH_CAR_OFFSET_1L to be equal to ZERO_LOW.

Save the file and re-build SD2 SW.

Example

Let it be VAL equal to 3342 (decimal).

Then, OFFSET =0xFFFBB282, ZERO_HIGH=dec(FFFB)=65531, ZERO_LOW=dec(B282)=45698.

The file bh_const.scr should be updated according to the following:

45698   EQU   BH_CAR_OFFSET_1L

65531   EQU   BH_CAR_OFFSET_1H

6. SD2 OPERATIONS: EXAMPLE

In the following, a typical drilling operation of SD2 is reported. It is worth noticing that before the execution of this command sequence, the COSAC PTOLEMY and LANDG backup RAMs shall properly initialized.

Starting drill translation position: 30 [1/100mm]; starting carousel position: 0 [arcmin]

As a reference it is assumed a distance of 200mm from the material (to include the effects of a non perfectly flat surface) and a sample at 310mm below the balcony.

The reference sequence is the following:

1. Switch on electronic

2. Translation down of tool till 280mm, 8mm/min, 375mA

3. Drilling:

- Translation down 2mm/min, 375mA to 390mm

- Drill rotation 704rpm (motor), 1.25A

till defined drilling quote

4. Translation up of 1mm, 2mm/min, 375mA

5. Sampling Tube release

6. Drill rotation for 20 sec, 704rpm (motor), 1.25A

7. Translation up of tool till 0.3mm, 8mm/min, 375mA

8. Carousel rotation (the oven 3 has been assumed to be the empty oven to put under drill axis), 1rpm, 600mA

9. Sample discharge: translation down 2mm/min, 375mA (for about 5 minutes)

10. Translation up of tool till 0.3mm, 2mm/min, 375mA

11. Carousel rotation (full oven to vol. ch. position), 1rpm, 600mA

12. Volume Checker down (80rpm, 450mA) and up (280rpm, 600mA) (total time: few seconds)

13. Carousel rotation (re-arm position), 1rpm, 600mA

14. Re-arm: Translation down 2mm/min, 375mA (for about 8 minutes)

15. Translation up of tool till 0.3mm, 2mm/min, 375mA
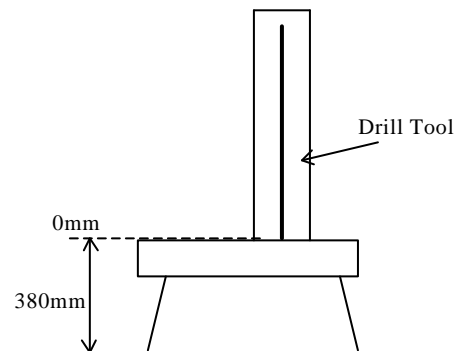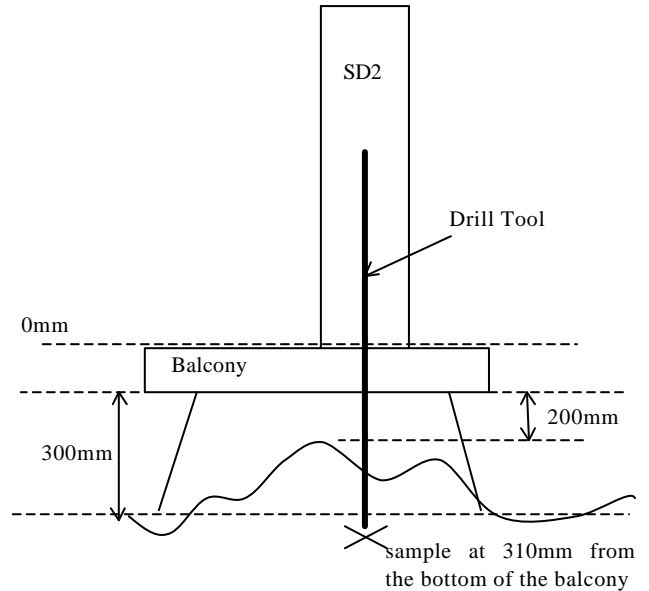
Operation Description	Command sequence (hex format)
Read Resolvers	0960 0960
Delay 8s	ABEF 0020 AC0F
Dump scientific data	8800 8800
Power off resolvers	0900 0900
Send LANDG data:	

Index 0 ranges [1 100] [200 300] [1000 2000] [10000 20000]	B0E0 0001 0064 00C8 012C 03E8 07D0 2710 4E20 3421
Step 2. Transl. Down to 280 mm, 503rpm, 375mA Acquire resolver Delay 8 s Dump scientific data Power off resolvers	2A72 6D60 97D2 0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
Step 3. Rotation to 704rpm, 1.25A, direction CW Transl. Down to 390 mm, 125rpm, 375mA stop rotation (at the end of translation) Acquire resolver Delay 8 s Dump scientific data Power off resolvers	3250 FFFF 324F 2972 9858 C1CA 3955 3955 0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
Step 4. Transl. Up to 389 mm, 125rpm, 375mA Acquire resolver Delay 8s Dump scientific data Power off resolvers	2972 97F4 C166 0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
Step 5. Sampling Tube release	6BB8 6BB8
Step 6. Rotation to 704rpm, 1.25A, t=20s wait for the end-of-rotation: delay 40 s	3250 0014 3264 ABEF 00A0 AC8F
Step 7. Transl. Up to 0.3mm, 503rpm, 375mA Acquire resolver Delay 8s Dump scientific data Power off resolvers	2A72 001E 2A90 0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
Step 8. Oven at position 3 under drill: carousel to 18720arcmin, 100rpm, 600mA Acquire resolver Delay 8s Dump scientific data Power off resolvers	193C 4920 0014 6270 193C 4920 0014 6270 0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
Step 9: discharge in the Oven: drill translation to 10 mm, 125mA, 375rpm Acquire resolver Delay 8 s Dump scientific data Power off resolvers	2972 03E8 2D5A 0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
Step 10. Transl. Up to 0.3mm, 125rpm, 375mA Acquire resolver Delay 8s Dump scientific data Power off resolvers	2972 001E 2990 0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
Step 11. Oven position 3 under volume checker: carousel to 17280 arcmin, 600mA, 100rpm Acquire resolver Delay 8s Dump scientific data Power off resolvers	193C 4380 0014 5CD0 193C 4380 0014 5CD0 0960 0960 ABEF 0020 AC0F 8800 8800 0900 0900
Step 12. volume checker Down 450mA, 80rpm volume checker Up 600mA, 280rpm	4095 00A0 4135 411F 0078 4197

dump scientific data	8800 8800
Step 13: Dummy Oven under Drill: carousel to 1440 arcmin, 600mA, 100rpm	193C 05A0 0014 1EF0 193C 05A0 0014 1EF0
Acquire resolver	0960 0960
Delay 8s	ABEF 0020 AC0F
Dump scientific data	8800 8800
Power off resolvers	0900 0900
Step 14: Re-Arm: drill translation to 15,4 mm, 375mA, 125rpm	2972 0604 2F76
Acquire resolver	0960 0960
Delay 8s	ABEF 0020 AC0F
Dump scientific data	8800 8800
Power off resolvers	0900 0900
Step 15 Transl. Up to 0.3mm, 125rpm, 375mA	2A72 001E 2A90
Acquire resolver	0960 0960
Delay 8s	ABEF 0020 AC0F
Dump scientific data	8800 8800
Power off resolvers	0900 0900
Carousel Zeroing 600mA, 100rpm	193C 0000 0014 1950
Acquire resolver	0960 0960
Delay 8s	ABEF 0020 AC0F
Dump scientific data	8800 8800
Power off resolvers	0900 0900

APPENDIX A: ERROR MESSAGES

In this chapter, for each failure the following information items are reported:

- Id: failure identifier (Hex) as reported in telemetry data
- mnemonics: the failure name used in this document and in [CDMSICD]
- error reason: why the failure is happens
- recovery procedure: how to remove the failure

Refer also to chapters "SD2 Checks", "SD2 Recovery Actions", "How to disable the SD2 Recovery Actions".

Error Id

0x0016: **mnemonics:** EC_SPC_CMD_DATA_OUT_OF_RANGE
error reason: some of the data specified in the command are out of the allowed range
recovery procedure: check that the command data are according to [CDMSICD], or check the communication CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0026: **mnemonics:** EC_SPC_CMD_CHECKSUM_FAILURE
error reason: command checksum is not correct
recovery procedure: check that the command checksum is correctly generated, or check the communication CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0036: **mnemonics:** EC_SPC_CMD_SYNTAX_FAILURE
error reason: the syntax of the command is not according to [CDMSICD]
recovery procedure: check that the command is according to [CDMSICD], or check the communication with CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0046: **mnemonics:** EC_SPC_CMD_ID_FAILURE
error reason: the fixed bits of the first command word (containing the specific command id) are not according to [CDMSICD]
recovery procedure: check that the command is according to [CDMSICD], or check the communication with CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0056: **mnemonics:** EC_SPC_CMD_WRDC_FAILURE
error reason: the number of words of the specific command is not correct
recovery procedure: check that the command is according to [CDMSICD], or check the communication with CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0066: **mnemonics:** EC_SPC_CMD_COMMAND_EMPTY
error reason: the command ACRE has both R1 and R2 bits equal to zero
recovery procedure: set to 1 the bit relevant to the resolver to be acquired. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0076: **mnemonics:** EC_SPC_CMD_NOT_ALLOWED_IN_MP
error reason: the command is not allowed within a mission plan

recovery procedure: do not use the command in a mission plan. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0081:   **mnemonics:** EC_SPC_CMD_REJ_DURING_LDMP

       error reason: the issued command can not be sent during the loading of a mission plan

       recovery procedure: If you want to execute the rejected command:

- either wait for the loading completion; the command will be executed at the end of the whole mission plan execution
- or interrupt the loading by means of EMST/ ABRT commands, send ENEM and then send the command

0x0091:   **mnemonics:** EC_SPC_CMD_REJECTED_IN_DEAD

       error reason: the command is rejected because SD2 is in "dead" status

       recovery procedure: check the SD2 status by issuing the command MHIT and analyze the telemetry contents. Issue ENEM in order to go to "ready" status

0x00A1:   **mnemonics:** EC_SPC_CMD_REJECTED_IN_UNDEFINED

       error reason: the command is rejected because SD2 is in "undefined" status

       recovery procedure: send the command STARTOP-SD2 Operation in order to go to "ready" status and then send the command again

0x00B6:   **mnemonics:** EC_SPC_CMD_REJECTED_IN_POWER_ON

       error reason: the command STOPOP-SD2 operation is rejected because some drivers are not powered off

       recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. Send the command ONOFF with all parameters equal to 0, and then send STOPOP-SD2 operation

0x00C6:   **mnemonics:** EC_ABRT_CMD_CHECKSUM_FAILURE

       error reason: command checksum of ABRT is not correct

       recovery procedure: check that the command checksum is correctly generated, or check the communication CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x00D6:   **mnemonics:** EC_EMST_CMD_CHECKSUM_FAILURE

       error reason: command checksum of EMST is not correct

       recovery procedure: check that the command checksum is correctly generated, or check the communication CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x00E6:   **mnemonics:** EC_ABRT_CMD_SYNTAX_FAILURE

       error reason: the syntax of the command ABRT is not according to [CDMSICD]

       recovery procedure: check that the ABRT command is according to [CDMSICD], or check the communication with CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x00F6:   **mnemonics:** EC_EMST_CMD_SYNTAX_FAILURE

       error reason: the syntax of the command EMST is not according to [CDMSICD]

       recovery procedure: check that the EMST command is according to [CDMSICD], or check the communication with CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

TECNOSPAZIO S.p.A   DOC. N°:SD2-SUM-TS-011
Date: October 2002   Revision : E
Project: SHARK   Sheet N°: 63

0x0106:   **mnemonics:** EC_ABRT_CMD_WRDC_FAILURE
     error reason: the number of words of the ABRT command is not correct
     recovery procedure: check that the ABRT command is according to [CDMSICD], or check the communication with CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0206:   **mnemonics:** EC_EMST_CMD_WRDC_FAILURE
     error reason: the number of words of the EMST command is not correct
     recovery procedure: check that the EMST command is according to [CDMSICD], or check the communication with CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0200:   **mnemonics:** EC_RERC_DESTINATION_UNIT_OFF
     error reason: the standard command with action code RERC and reason destination-unit-off has been received by SD2
     recovery procedure: nothing

0x0215:   **mnemonics:** EC_RERC_SCIENCE_DATA_OUT
     error reason: the standard command with action code RERC and reason science-data-out has been received by SD2
     recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status

0x0225:   **mnemonics:** EC_RERC_MEMORY_FULL
     error reason: the standard command with action code RERC and reason memory-full has been received by SD2
     recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status

0x0235:   **mnemonics:** EC_RERC_REQ_UNDUE
     error reason: the standard command with action code RERC and reason request-undue has been received by SD2
     recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status

0x0245:   **mnemonics:** EC_RERC_ILLEGAL_ADDRESS
     error reason: the standard command with action code RERC and reason illegal-address has been received by SD2
     recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status

0x0255:   **mnemonics:** EC_RERC_ILLEGAL_REQ_CODE
     error reason: the standard command with action code RERC and reason illegal-request-code has been received by SD2
     recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status

0x0261:   **mnemonics:** EC_RERC_REASON_UNKNOWN
     error reason: the standard command with action code RERC and reason unknown has been received by SD2
     recovery procedure: nothing

0x0270:   **mnemonics:** EC_RASV_RECEIVED
     error reason: the standard command with action code RASV has been received by SD2

recovery procedure: nothing

0x0280:   **mnemonics:** EC_RAXT_RECEIVED
       error reason: the standard command with action code RAXT has been received by SD2
       recovery procedure: nothing

0x0290:   **mnemonics:** EC_RSCS_RECEIVED
       error reason: the standard command with action code RSCS has been received by SD2
       recovery procedure: nothing

0x02A0:   **mnemonics:** EC_RBUS_RECEIVED
       error reason: the standard command with action code RBUS has been received by SD2
       recovery procedure: nothing

0x02B1:   **mnemonics:** EC_TRG_RECEIVED
       error reason: the standard command with action code TTRG or RTRG has been received by SD2
       recovery procedure: nothing

0x02D5:   **mnemonics:** EC_USR_CMD_ILLEGAL_AC
       error reason: The action code received from the USR$CMD routine is not within the allowed range of CDMS.
       recovery procedure: check that the action codes are according to [CDMSICD], or check the communication CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x02E5:   **mnemonics:** EC_USR_DATA_ILLEGAL_AC
       error reason: The action code received from the USR$DATA routine is not within the allowed range of CDMS.
       recovery procedure: check that the action codes are according to [CDMSICD], or check the communication CDMS- SD2. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x02F5:   **mnemonics:** EC_STBY_RECEIVED
       error reason: the standard command with action code STBY has been received by SD2
       recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status

0x0305:   **mnemonics:** EC_RCMS_RECEIVED_WITH_NO_MP_LOAD
       error reason: Problem in CDMS-SD2 communication
       recovery procedure: No recovery action is foreseen: the error shall be notified to TS by means of SPR. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0311:   **mnemonics:** EC_RMOD_NORMAL_MODE
       error reason: the standard command with action code RMOD has been received by SD2 with NORMAL MODE option.
       recovery procedure: nothing

0x0321:   **mnemonics:** EC_RMOD_NO_NORMAL_MODE
       error reason: the standard command with action code RMOD has been received by SD2 with NO NORMAL MODE option.
       recovery procedure: nothing

0x0331: **mnemonics:** EC_RSST_INIT_RECOVERY_PROC
error reason: the standard command with action code RSST has been received by SD2 with INITIATE RECOVERY PROCESS option.
recovery procedure: nothing

0x0340: **mnemonics:** EC_RSST_NO_RECOVERY_PROC
error reason: the standard command with action code RSST has been received by SD2 with NO INITIATE RECOVERY PROCESS option.
recovery procedure: nothing

0x0403: **mnemonics:** EC_CHK_DRILLROT_SPEED_FAILURE
error reason: see chapter "Drill Rotation Speed Check"
recovery procedure: see chapter "Drill Rotation Speed Check". In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. This error shall be notified to TS by means of SPR.

0x0412: **mnemonics:** EC_CHK_DRILLTRA_SPEED_FAILURE
error reason: see chapter "Drill Translation Speed Check"
recovery procedure: see chapter "Drill Translation Speed Check". This error shall be notified to TS by means of SPR.

0x0423: **mnemonics:** EC_CHK_CAROUSEL_SPEED_FAILURE
error reason: see chapter "Carousel Speed Check"
recovery procedure: see chapter "Carousel Speed Check". In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. This error shall be notified to TS by means of SPR.

0x0506: **mnemonics:** EC_BH_SWITCHONOFF_ERROR
error reason: SW bug.
recovery procedure: No recovery action is foreseen: the error shall be notified to TS by means of SPR. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0516: **mnemonics:** EC_BH_SETSTD_ERROR
error reason: SW bug.
recovery procedure: No recovery action is foreseen: the error shall be notified to TS by means of SPR. In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue.

0x0526: **mnemonics:** EC_BH_SWITCHTUBE_ERROR
error reason: HW problem
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. This error shall be notified to TS by means of SPR.

0x0603: **mnemonics:** EC_DEADLINE_MISSED
error reason: HW-SW problem: all the HW-SW operations scheduled to be completed within 250 [ms] require more than 250 [ms].
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. This error shall be notified to TS by means of SPR.

0x0613: **mnemonics:** EC_COSAC_STATUS_ENGAGED

error reason: The COSAC tapping station status has been retrieved via COSAC backup RAM reading. Its contents is "engaged". See chapter "Tapping Station Status Check".
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. Be sure that COSAC tapping station is really disengaged before to command the carousel motion.

0x0623:   **mnemonics:** EC_COSAC_STATUS_UNDEFINED
error reason: The COSAC tapping station status has been retrieved via COSAC backup RAM reading. Its contents is "undefined". See chapter "Tapping Station Status Check".
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. Be sure that COSAC tapping station is really disengaged before to command the carousel motion.

0x0633:   **mnemonics:** EC_PTOLEMY_STATUS_ENGAGED
error reason: The PTOLEMY tapping station status has been retrieved via PTOLEMY backup RAM reading. Its contents is "undefined". See chapter "Tapping Station Status Check".
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. Be sure that PTOLEMY tapping station is really disengaged before to command the carousel motion.

0x0643:   **mnemonics:** EC_PTOLEMY_STATUS_UNDEFINED
error reason: The PTOLEMY tapping station status has been retrieved via PTOLEMY backup RAM reading. Its contents is "undefined". See chapter "Tapping Station Status Check".
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. Be sure that PTOLEMY tapping station is really disengaged before to command the carousel motion.

0x0656:   **mnemonics:** EC_SARE_CONFIGURATION_CHECK_FAILURE
error reason: Either the carousel position or the drill translation position does not allow the "sampling tube release" command- SARE command. See chapter "SD2 Mechanical Correctness Checks".
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Acquire the carousel and the drill translation position (you can make use of the "SD2 Health Check" procedure of "SD2 Subsystem Limited Functional Tests- Inputs" document) and check if these positions are the foreseen ones. If not so, notify the failure to TS by means of SPR. If they are the foreseen carousel/ drill translation positions, command either the carousel to zero or the drill translation to zero.

0x0664:   **mnemonics:** EC_DRILL_POS_CHECK_FAILURE
error reason: The drill translation position does not allow the carousel motion. See chapter "SD2 Mechanical Correctness Checks".
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Acquire the carousel and the drill translation position (you can make use of the "SD2 Health Check" procedure of "SD2 Subsystem Limited Functional Tests- Inputs" document) and check if these positions are the foreseen ones. If not so, notify the failure to TS by means of SPR. If they are the foreseen carousel/ drill translation positions, command the drill translation to move in a position allowing the carousel motion.

0x0674:   **mnemonics:** EC_RESOLVER_DATA_INVALID
error reason: HW-SW error: problem during resolver reading
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x0684: **mnemonics:** EC_CAROUSEL_POS_CHECK_FAILURE
error reason: the carousel position does not allow the drill translation motion or the volume checker motion in direction "down". See chapter "SD2 Mechanical Correctness Checks".
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Acquire the carousel and the drill translation position (you can make use of the "SD2 Health Check" procedure of "SD2 Subsystem Limited Functional Tests- Inputs" document) and check if these positions are the foreseen ones. If not so, notify the failure to TS by means of SPR. If they are the foreseen carousel/ drill translation positions, command the carousel to move in a position allowing the drill translation or the volume checker motion.

0x0693: **mnemonics:** EC_VCK_MICROSW_CHECK_FAILURE
error reason: see chapter "Volume Checker Microswitch Check".
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR. Send again the MVCK/ VCAC command by using a torque value higher than the value used in the command causing the error

0x06A3: **mnemonics:** EC_VOLUME_COUNTER_INVALID
error reason: HW-SW error: problem during volume checker counter reading
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x06B3: **mnemonics:** EC_LANDG_POS_CHECK_FAILURE
error reason: The landing gear position has been retrieved via LANDG backup RAM reading: it does not allow moving the drill translation. See chapter "Landing Gear Position Check".
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. Be sure that LANDING gear position allows the drill translation motion before to issue the drill translation command again.

0x06C3: **mnemonics:** EC_DRTR_MAIN_MOTION_CHECK_FAILURE
error reason: the check on drill translation main winding is failed. See chapter "Drill Translation Windings Check".
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. See chapter "Drill Translation Windings Check" for failure meaning. Notify the failure to TS by means of SPR.

0x06D3: **mnemonics:** EC_DRTR_RED_MOTION_CHECK_FAILURE
error reason: the check on drill translation redundant winding is failed. See chapter "Drill Translation Windings Check".
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. See chapter "Drill Translation Windings Check" for failure meaning. Notify the failure to TS by means of SPR

0x0716: **mnemonics:** EC_PH_HW_ERROR
error reason: HW-SW error: problem during access to drill translation/ carousel boards
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x0726: **mnemonics:** EC_PH_POSE_OUT_OF_RANGE
error reason: the commanded position overcomes the allowed upper values: allowed range for the carousel and for the drill translation is reported in [CDMSICD].
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status.

0x0733: **mnemonics:** EC_PH_CAR_RAMP_DOWN_TIMEOUT
error reason: HW-SW error: the ramp-down trajectory of carousel (see [URD]) is not completed within the C_CAR_RAMP_DOWN_MAX_DURATION time (as specified in chapter "SD2 SW constants").
recovery procedure: In this situation SD2 enters in "dead" status; send ENEM command to go to "ready" status and to continue. See chapter "Drill Translation Windings Check" for failure meaning. Notify the failure to TS by means of SPR.

0x0806: **mnemonics:** EC_SH_HW_ERROR
error reason: HW-SW error: problem during access to drill rotation/ volume checker boards
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR.

0x1006: **mnemonics:** EC_Q_SH_FULL_QUEUE
error reason: SW error: the queue of messages related to SH module is full.
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1016: **mnemonics:** EC_Q_EH_FULL_QUEUE
error reason: SW error: the queue of messages related to EH module is full.
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1026: **mnemonics:** EC_Q_TM_FULL_QUEUE
error reason: SW error: the queue of messages related to TM module is full.
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1036: **mnemonics:** EC_Q_ELPTIM_FULL_QUEUE
error reason: SW error: the queue of messages related to C_TIMER module is full.
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1046: **mnemonics:** EC_Q_EMSCMD_FULL_QUEUE
error reason: SW error: the queue of messages related to EMERGCY module is full.
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1056: **mnemonics:** EC_Q_CDMS_FULL_QUEUE
error reason: SW error: the queue of messages related to CDMSIFHL module is full.
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1066: **mnemonics:** EC_Q_TIMER_FULL_QUEUE
error reason: SW error: the queue of messages related to C_TIMER module is full.
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1076: **mnemonics:** EC_Q_EM_CIH_FULL_QUEUE

error reason: SW error: the queue of messages sent to the CDMSIFHL module is full.
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1086: **mnemonics:** EC_Q_PH_FULL_QUEUE
error reason: SW error: the queue of messages related to PH module is full
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1096: **mnemonics:** EC_Q_SCMD_FULL_QUEUE
error reason: SW error: the queue of messages related to SCMDHNL module is full
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x10A6: **mnemonics:** EC_Q_CHK_FULL_QUEUE
error reason: SW error: the queue of messages related to C_CHK module is full
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1106: **mnemonics:** EC_Q_RH_FULL_QUEUE
error reason: SW error: the queue of messages related to C_RH module is full
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x1116: **mnemonics:** EC_Q_RERC_FULL_QUEUE
error reason: SW error: the queue of messages related to C_RH module is full
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2006: **mnemonics:** EC_SH_INTERNAL_ERROR
error reason: SW error in SH module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2016: **mnemonics:** EC_TM_INTERNAL_ERROR
error reason: SW error in C_TELEM module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2026: **mnemonics**: EC_SCMD_INTERNAL_ERROR
error reason: SW error in SCMDHNL module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2036: **mnemonics:** EC_SCMDLIB_INTERNAL_ERROR
error reason: SW error in SCMD_LIB module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2046: **mnemonics:** EC_BITLIB_INTERNAL_ERROR

error reason: SW error in BITLIB module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2056: **mnemonics:** EC_CIH_INTERNAL_ERROR
error reason: SW error in CDMSIFHL module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2066: **mnemonics:** EC_OPRT_INTERNAL_ERROR
error reason: SW error in OPRT module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2076: **mnemonics:** EC_TIMER_INTERNAL_ERROR
error reason: SW error in C_TIMER module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2086: **mnemonics:** EC_MVCK_INTERNAL_ERROR
error reason: SW error in MVCK module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2096: **mnemonics:** EC_GBUS_IO_INTERNAL_ERROR
error reason: SW error in GBUS_IO module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x20A6: **mnemonics:** EC_DIRECT_INTERNAL_ERROR
error reason: SW error in DIRECT module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x20B6: **mnemonics:** EC_PH_INTERNAL_ERROR
error reason: SW error in PH module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x20C6: **mnemonics:** EC_CHK_INTERNAL_ERROR
error reason: SW error in C_CHK module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x20D6: **mnemonics:** EC_CAPO_INTERNAL_ERROR
error reason: SW error in CAR module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x20E6: **mnemonics:** EC_DRILL_INTERNAL_ERROR

error reason: SW error in DRILL module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x20F6:   **mnemonics:** EC_EMERGCY_INTERNAL_ERROR

error reason: SW error in EMERGCY module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2106:   **mnemonics:** EC_Q_SCMD_INTERNAL_ERROR

error reason: SW error in Q_SCMD module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2116:   **mnemonics:** EC_Q_CDMSIF_INTERNAL_ERROR

error reason: SW error in Q_CDMSIF module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2126:   **mnemonics:** EC_ZERO_INTERNAL_ERROR

error reason: SW error in ZERO module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2136:   **mnemonics:** EC_VCAC_INTERNAL_ERROR

error reason: SW error in VCAC module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2146:   **mnemonics:** EC_RH_INTERNAL_ERROR

error reason: SW error in C_RH module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2156:   **mnemonics:** EC_RH_LIB_INTERNAL_ERROR

error reason: SW error in RH_LIB module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2166:   **mnemonics:** EC_BRR_INTERNAL_ERROR

error reason: SW error in BRR module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2176:   **mnemonics:** EC_MPL_INTERNAL_ERROR

error reason: SW error in MPLOADER module

recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2186:   **mnemonics:** EC_BH_INTERNAL_ERROR

error reason: SW error in BH module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x2196: **mnemonics:** EC_SCHEDUL_INTERNAL_ERROR
error reason: SW error in SCHEDUL module
recovery procedure: SD2 enters in "dead" status; send ENEM command to go to "ready" status. Notify the failure to TS by means of SPR

0x3001: **mnemonics:** EC_LDMP_REJECTED_IN_DEAD
error reason: The specific command LDMP has been rejected (so that no mission plan is loaded) since SD2 is in DEAD status
recovery procedure: send at first specific command ENEM to exit from emergency; then, send again the LDMP specific command

0x3011: **mnemonics:** EC_LDMP_REJECTED_IN_UNDEF
error reason: The specific command LDMP has been rejected (so that no mission plan is loaded) since SD2 is in UNDEFINED status
recovery procedure: send at first the specific command STARTOP with parameter SD2-operation; then, send again LDMP specific command

0x3021: **mnemonics:** EC_LDMP_REJ_DURING_LDMP
error reason: The specific command LDMP has been rejected (so that no new mission plan is loaded) since another load-mission-plan is in progress
recovery procedure: wait for the end of previous load-mission-plan, and then send again LDMP specific command

0x3031: **mnemonics:** EC_LDMP_REJECTED_IN_DUMP
error reason: The specific command LDMP has been rejected (so that no mission plan is loaded) since there is a scientific data dumping in progress
recovery procedure: wait for the end of scientific data dumping, and then send again LDMP specific command

0x3041: **mnemonics:** EC_LDMP_CMD_CHECKSUM_FAILURE
error reason: LDMP command checksum is not correct (so that no mission plan is loaded)
recovery procedure: check that the command checksum is correctly generated, or check the communication CDMS- SD2. Then send again the correct LDMP command

0x3051: **mnemonics:** EC_LDMP_SSCMD1_SYNTAX_ERR
error reason: The syntax of first word of LDMP command is not correct (so that no mission plan is loaded)
recovery procedure: check the syntax of first word of LDMP command, or check the communication CDMS- SD2. Then send again the correct LDMP command

0x3061: **mnemonics:** EC_LDMP_OFFSET_PARAM_ERR
error reason: The OFFSET parameter of LDMP specific command is not within the allowed range specified in [CDMSICD] (so that no mission plan is loaded)
recovery procedure: check the OFFSET parameter of LDMP command, or check the communication CDMS- SD2. Then send again the correct LDMP command.

0x3071: **mnemonics:** EC_LDMP_LEN_PARAM_ERR

error reason: The LENGTH parameter of LDMP specific command is not within the allowed range specified in [CDMSICD] (so that no mission plan is loaded)

recovery procedure: check the LENGTH parameter of LDMP command, or check the communication CDMS- SD2. Then send again the correct LDMP command

0x3081:   **mnemonics:** EC_LDMP_OFF_LEN_PARAM_ERR

error reason: The sum of OFFSET and LENGTH parameters of LDMP specific command is not within the allowed range specified in [CDMSICD] (so that no mission plan is loaded)

recovery procedure: check the LENGTH and OFFSET parameters of LDMP command, or check the communication CDMS- SD2. Then send again the correct LDMP command

0x3091:   **mnemonics:** EC_MP_ADLER32_CKSUM_FAILURE

error reason: The Adler32 checksum (parameters S1 S2 of LDMP command) is not correct. In this case, SD2 automatically retries the loading. As a maximum, SD2 performs three trials. If at the end of three trials the Adler32 checksum is not correct, the error EC_MP_LOAD_FAILED is raised (so that no mission plan is loaded)

recovery procedure: check the S1 S2 parameters of LDMP command, or check the communication CDMS- SD2.

0x30A1:   **mnemonics:** EC_LDMP_CMD_WRDC_FAILURE

error reason: the number of words of the specific command is not correct (so that no mission plan is loaded)

recovery procedure: check that the LDMP command is according to [CDMSICD], or check the communication with CDMS- SD2. Then send again the correct LDMP command.

0x30B1:   **mnemonics:** EC_MP_LOAD_FAILED

error reason: this error is raised when SD2 detects the EC_MP_ADLER32_CKSUM_FAILURE for three times on the loaded mission plan. In this case, the loaded mission plan is cleared.

recovery procedure: check the S1 S2 parameters of LDMP command, or check the communication CDMS- SD2. Then send again the correct LDMP command

0x4001:   **mnemonics:** EC_RERC_DUO_RECEIVED_WITHOUT_REQ

error reason: The message RERC (Receive Error Code Word) is received from CDMS with reason "Destination Unit Off" while there is no pending request to CDMS

recovery procedure: check communication SD2- CDMS

0x4011:   **mnemonics:** EC_RERC_SDO_RECEIVED_WITHOUT_REQ

error reason: The message RERC (Receive Error Code Word) is received from CDMS with reason "Science Data Volume Exhausted" while there is no pending request to CDMS

recovery procedure: check communication SD2- CDMS

0x4021:   **mnemonics:** EC_RERC_MF_RECEIVED_WITHOUT_REQ

error reason: The message RERC (Receive Error Code Word) is received from CDMS with reason "Mass Memory Full" while there is no pending request to CDMS

recovery procedure: check communication SD2- CDMS

0x4031:   **mnemonics:** EC_RERC_RU_RECEIVED_WITHOUT_REQ

error reason: The message RERC (Receive Error Code Word) is received from CDMS with reason "Request Undue" while there is no pending request to CDMS

recovery procedure: check communication SD2- CDMS

0x4041: **mnemonics:** EC_RERC_IA_RECEIVED_WITHOUT_REQ
    error reason: The message RERC (Receive Error Code Word) is received from CDMS with reason "Illegal pointer, offset, number" while there is no pending request to CDMS
    recovery procedure: check communication SD2- CDMS

0x4051: **mnemonics:** EC_RERC_IR_RECEIVED_WITHOUT_REQ
    error reason: The message RERC (Receive Error Code Word) is received from CDMS with reason "Illegal Request Code" while there is no pending request to CDMS
    recovery procedure: check communication SD2- CDMS

0x4071: **mnemonics:** EC_SRDY_TIMEOUT_EXPIRED
    error reason: a timeout occurs while SD2 is waiting for the TRQC by CDMS in order to send the SRDY (Science Data Ready) request. When this timeout occurs, SD2 automatically re-sets the SR (Service Request Flag) and waits again for the TRQC. This for C_CDMS_NUM_RETRY times. The timeout value (C_CDMS_TIMEOUT) and the constant value of C_CDMS_NUM_RETRY are reported in chapter "SD2 SW Constants"
    recovery procedure: check communication SD2- CDMS

0x4081: **mnemonics:** EC_FLSP_TIMEOUT_EXPIRED
    error reason: a timeout occurs while SD2 is waiting for the TRQC by CDMS in order to send the FLSP (Flush Last Science Data Packet) request. When this timeout occurs, SD2 automatically re-sets the SR (Service Request Flag) and waits again for the TRQC. This for C_CDMS_NUM_RETRY times. The timeout value (C_CDMS_TIMEOUT) and the constant value of C_CDMS_NUM_RETRY are reported in chapter "SD2 SW Constants"
    recovery procedure: check communication SD2- CDMS

0x4091: **mnemonics:** EC_OCPL_TIMEOUT_EXPIRED
    error reason: a timeout occurs while SD2 is waiting for the TRQC by CDMS in order to send the OCPL (Operation Completed) request. When this timeout occurs, SD2 automatically re-sets the SR (Service Request Flag) and waits again for the TRQC. This for C_CDMS_NUM_RETRY times. The timeout value (C_CDMS_TIMEOUT) and the constant value of C_CDMS_NUM_RETRY are reported in chapter "SD2 SW Constants"
    recovery procedure: check communication SD2- CDMS

0x40A1: **mnemonics:** EC_SCMD_TIMEOUT_EXPIRED
    error reason: a timeout occurs while SD2 is waiting for the TRQC by CDMS in order to send the SCMD (Send Stored Tcmd Buffer Section) request. When this timeout occurs, SD2 automatically re-sets the SR (Service Request Flag) and waits again for the TRQC. This for C_CDMS_NUM_RETRY times. The timeout value (C_CDMS_TIMEOUT) and the constant value of C_CDMS_NUM_RETRY are reported in chapter "SD2 SW Constants"
    recovery procedure: check communication SD2- CDMS

0x40B1: **mnemonics:** EC_RDBF_TIMEOUT_EXPIRED
    error reason: a timeout occurs while SD2 is waiting for the TRQC by CDMS in order to send the RDBF (Read Backup RAM Buffer Record) request. When this timeout occurs, SD2 automatically re-sets the SR (Service Request Flag) and waits again for the TRQC. This for C_CDMS_NUM_RETRY times. The timeout value (C_CDMS_TIMEOUT) and the constant value of C_CDMS_NUM_RETRY are reported in chapter "SD2 SW Constants"
    recovery procedure: check communication SD2- CDMS

0x40C1: **mnemonics:** EC_WRBF_TIMEOUT_EXPIRED

error reason: a timeout occurs while SD2 is waiting for the TRQC by CDMS in order to send the WRBF (Write Backup RAM Buffer Record) request. When this timeout occurs, SD2 automatically re-sets the SR (Service Request Flag) and waits again for the TRQC. This for C_CDMS_NUM_RETRY times. The timeout value (C_CDMS_TIMEOUT) and the constant value of C_CDMS_NUM_RETRY are reported in chapter "SD2 SW Constants"
recovery procedure: check communication SD2- CDMS

0x40D1:   **mnemonics:** EC_SRDY_MAX_NUM_RETRY_REACHED
error reason: SD2 is trying to send the SRDY (Science Data Ready) request but a timeout occurs C_CDMS_NUM_RETRY times. SD2 removes the current SRDY request so that the buffer of scientific data that should be dumped by means of the removed SRDY is lost.
recovery procedure: check communication SD2- CDMS.

0x40E1:   **mnemonics:** EC_FLSP_MAX_NUM_RETRY_REACHED
error reason: SD2 is trying to send the FLSP (Flush Last Science Data Packet) request but a timeout occurs C_CDMS_NUM_RETRY times. SD2 removes the current FLSP request so that the buffer of scientific data that should be dumped by means of the removed FLSP is lost.
recovery procedure: check communication SD2- CDMS.

0x40F3:   **mnemonics:** EC_OCPL_MAX_NUM_RETRY_REACHED
error reason: SD2 is trying to send the OCPL (Operation Completed) request but a timeout occurs C_CDMS_NUM_RETRY times. SD2 removes the current OCPL request so that the "Operation Completed" is not notified to CDMS.
recovery procedure: check communication SD2- CDMS.

0x4101:   **mnemonics:** EC_SCMD_MAX_NUM_RETRY_REACHED
error reason: SD2 is trying to send the SCMD (Send Stored Tcmd Buffer Section) request but a timeout occurs C_CDMS_NUM_RETRY times. SD2 removes the current SCMD request so that the buffer section containing stored telecommand is not retrieved by SD2. Probably, at the end of loading the whole stored telecommands, Adler32 checksum error will be raised.
recovery procedure: check communication SD2- CDMS.

0x4113:   **mnemonics:** EC_RDBF_MAX_NUM_RETRY_REACHED
error reason: SD2 is trying to send the RDBF (Read Backup RAM Buffer Record) request but a timeout occurs C_CDMS_NUM_RETRY times. SD2 removes the current RDBF request so that SD2 does not get the current status of requested backup RAM.
recovery procedure: check communication SD2- CDMS. SD2 enters in "dead" status; send ENEM command to go to "ready" status.

0x4123:   **mnemonics:** EC_WRBF_MAX_NUM_RETRY_REACHED
error reason: SD2 is trying to send the WRBF (Write Backup RAM Buffer Record) request but a timeout occurs C_CDMS_NUM_RETRY times. SD2 removes the current WRBF request so that SD2 does not update the status of its backup RAM.
recovery procedure: check communication SD2- CDMS. SD2 enters in "dead" status; send ENEM command to go to "ready" status.