SUBJECT: **EGSE User Requirements**

PREPARED BY: **K.J. King**

DOCUMENT No: SPIRE-RAL-DOC-000102

ISSUE: Issue 1.0 **Date:** 17th December 2008

APPROVED BY: **Date:**

K.J. King (SPIRE)

Digitally signed by Ken King
DN: cn=Ken King, o=STFC, ou=RAL-SSTD, email=ken.king@stfc.ac.uk, c=US
Date: 2008.12.17 15:39:54 Z

Distribution

SPIRE

K.King	RAL
S Guest	RAL
J. Porter	RAL

Change Record

ISSUE	DATE	Changes
1.0	17th December 2008	First Issue as a SPIRE document in the light of no further comments from PACS and HIFI managers

TABLE OF CONTENTS

FIGURES

TABLES

Glossary

APID	Application ID
CDMS	Command and Data Management System
DDS	Data Distribution System
EGSE	Electrical Ground Support Equipment
ERT	Earth Received Time
FINDAS	FIRST Integrated Network and Data Archive System
FIRST	Far-Infrared and Sub-millimetre Telescope
HIFI	Heterodyne Instrument for FIrst
IA	Interactive Analysis
ICD	Interface Control Document
I/F	Interface
kbps	Kilo-bits per second
MIB	Mission Information Base
MOC	Mission Operations Centre
PACS	Photoconductor Array Camera and Spectrometer
PCS	Power Control System
QLA	Quick-Look Analysis
SPIRE	Spectral and Photometric Imaging Receiver
TEI	Test-Equipment Interface

A **command timeline** defines a sequence of commands to be sent to the spacecraft, with timing information that identifies at what time each command is to be sent. This time is specified relative to the beginning of the timeline. The timeline is expanded to a sequence of telecommand packets and either inserted into a schedule, during operations, or send to the spacecraft (or simulator) during ground testing. (Note: SCOS2000 documentation calls this a command sequence.)

1. INTRODUCTION

It has been agreed that an important contribution to minimising the resource requirements for the development and operation of the FIRST satellite is a common approach to instrument testing and in-flight operations, with the maximum reuse of equipment and software in all phases of the mission. In addition, it is advantageous for the three scientific instruments to collaborate on development of a common instrument test system (called the Electrical Ground Support Equipment, EGSE) to minimise the effort required, by removing duplication of work. Discussions have been held between the three FIRST instruments and ESA/ESOC with the intention of producing an agreed approach to the implementation of the EGSE. The results of these discussions and a conceptual design are documented in RD1.

1.1 Scope

This document provides the high-level requirements on the parts of the Electrical Ground Support Equipment as described in RD1 that are common to all the FIRST instruments. It does not address those parts of the EGSE specific to any one instrument.

At present no requirements documents are available for the full EGSE system. However, the IID Parts A and B (AD1 & AD2) (will) describe the interfaces between the spacecraft and the instruments and those between the EGSE and the system-level checkout equipment. The SIRD (AD3) places requirements on the software when used by the ICCs, during the operational phase. PACS has produced documents on the RTA requirements (RD1), ICC software requirements (RD2) and the ILT scenario (RD3) which contain relevant information.

1.2 Structure of Document

The next section of this document gives an overview of the EGSE system and describes its components. Section 3 describes the different environments within which the EGSE must operate when used in the different phases of the FIRST mission and identifies which components of the EGSE system are used at that time. Section 4 goes into more detail of the requirements on each component.

1.3 Documents

1.3.1 Applicable Documents

AD1:	Instrument Interface Document Part A	PT-IID-A-04624
AD2:	Instrument Interface Document Part B	
	HIFI	PT-HIFI-02125
	PACS	PT-PACS-02126
	SPIRE	PT-SPIRE-02124
AD3:	Science Implementation Requirements Document, SIRD	PT-03646
AD4:	EGSE Interface Unit Functional Description	SRON-U/HIFI/TN/2000-001

1.3.2 Reference Documents

RD1:	Common Instrument EGSE Concepts	FIRST-SPI-NOT-000097
RD2:	FIRST/Planck RTA URD, Issue 1 (4th November 1999)	
RD3:	PACS ICC Software URD	

RD4:	PACS ILT Scenario	
RD5:	Packet Structure ICD	SCI-PT-IF-07527
RD6:	RTA Delta URD, Issue 2 (L.Dubbeldam, 6th June 2000)	
RD7:	Additional RTA Requirements, (E.Wiezorrek, email 3rd April 2000),	
RD8:	MIB Import ICD, Issue 3.2 (M.Schick, A.Kowalczyk, A.Ercolani, 12th March 1999)	SCOS2000-ICD-MIBimport-3.2
RD9:	FIRST Common Uplink System (CUS) URD, Draft 1 (H.Feuchtgruber, U.Klaas)	PICC-ME-RD-001
RD10:	Comments on the CUS URD, (email from P.R.Roelfsema)	
RD11:	Baseline Requirements Document (for the ISO) Calibration Uplink System, (Version 1.7, 20-December-1993)	SAI/93-94/Dc
RD12:	Towards a Common Uplink System for FIRST, Discussion Paper for Concept Development Based on Elements of the ISO Uplink System, (Version 2.0, 28-June-1999)	PICC-MA-TN-001 (previously PACS-MA-TN-001)

2. EGSE OVERVIEW

The EGSE consists of several components, which together provide the functionality required to test and operate the FIRST instruments at various stages of their development. It is also expected that the majority of these components may be employed in the Ground Segment during the operational phase of the FIRST mission. This is achieved by implementing the EGSE system in two parts:

Part1 deals with the data handling and real-time interaction with the instrument, simulating the operations of the spacecraft and the satellite control centre. It is not needed during the operational phase. It consists of:

The **Interface Unit** that provides hardware interfaces to the instrument and the external test equipment required to stimulate the instrument during testing. It simulates the telecommand and telemetry interfaces of the spacecraft Command and Data Management Subsystem (CDMS), the spacecraft Power Control Subsystem (PCS)) and the thermometry interfaces. Control and monitoring of this unit is through the internal EGSE link with the Uplink and Downlink components. This interface is likely to be an ethernet connection using the TCP/IP protocol carrying raw telecommand and telemetry packets (the PDU simulator, thermometry and external test equipment will be treated as subsystems of the spacecraft as far as commanding and telemetry are concerned).

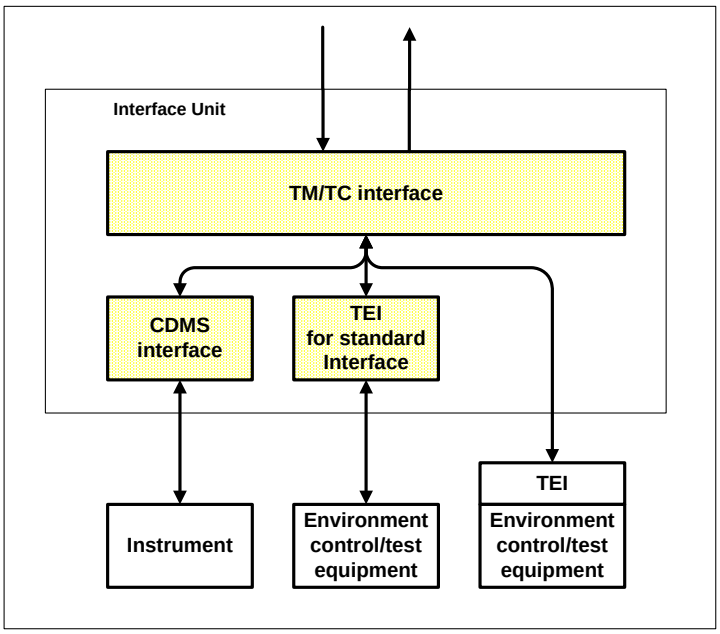

Figure 2-1 The Interface Unit

The structure of the Interface Unit is shown in figure 2-1. The TM/TC interface routes incoming packets to the appropriate interface: Telecommand packets addressed to the instrument will be passed directly to it through the CDMS interface, packets addressed to test equipment will be routed to an internal Test Equipment Interface (TEI) for standard interfaces (e.g. IEEE bus interface) or to the TEI of instrument specific test equipment The interface between the TM/TC interface and the TEIs will be a standarad TCP/IP network interface allowing instrument specific test equipment to be controlled independent of the EGSE, if required.

The **Test Control** component provides facilities for generating and uplinking commands (from command timelines, from interactive input from the user/operator in the form of mnemonics and parameters in engineering units and from OBS memory patches), executing test procedures and controlling operation of the system. This component also implements the real-time autonomy functions provided by the CDMS in flight. This means that it needs to provide facilities to respond to events generated by the RTA component, if necessary by taking appropriate action, including commanding the instrument and/or test equipment.

The **Uplink and Downlink components** provide functions to convert telecommand and telemetry packets between the format used by the Interface Unit and that provided to FINDAS (consolidated telemetry). The downlink component also provides absolute time stamping of the received telemetry packets, to allow correlation with other activities.

Part 2 provides facilities for storage, display and analysis of instrument data, maintenance of on-board software and generation of instrument commands. This part is used also during the operational phase. It consists of:

The **Real Time Assessment (RTA)** system, which extracts, converts and displays the housekeeping parameters contained in the telemetry stream, monitors these parameters against the expected value/status, performs command verification and on-board memory checking and monitors event information from the instrument and test equipment. It provides information for the Test Control component (e.g. parameter values) used to control test procedures and may generate events for immediate action by the Test Control component.

The **Common Uplink System** allows generation of command timelines (with command timing relative to the start of the timeline) from user input. Initially this will be from scripts describing the command sequences to be performed, later it will also allow input from Astronomical Observation Templates (AOTs) to be processed into timelines. This will need early definition of the format of command timelines.

The **Quick-Look Analysis (QLA) / Interactive Analysis (IA)** system will allow science data to be processed, displayed and analysed in near real time. It is expected that this subsystem will grow over time to provide all the analysis functions required in the operations phase.

The **On-Board Software Maintenance** component provides facilities to modify the instrument on-board software and generate code images and data in a format suitable for use by the Test Control component (or MOC in the operational environment) to uplink to the instrument and to allow checking of the on-board code.

These two parts are linked by the FINDAS component. This is a version of the database and communication software to be used in the FIRST Ground Segment. It provides facilities to store instrument data, test procedures, test logs and other information during the tests and provides access to this information at a later date. The intention is to provide a consistent interface to the second part of the EGSE through all stages of test and operations

3. OPERATIONAL ENVIRONMENTS

3.1 Instrument-level Tests

Figure 3-1 shows the EGSE configuration during these tests. Note that telecommands pass directly from the Test Control to the Uplink. All other data passes through FINDAS.

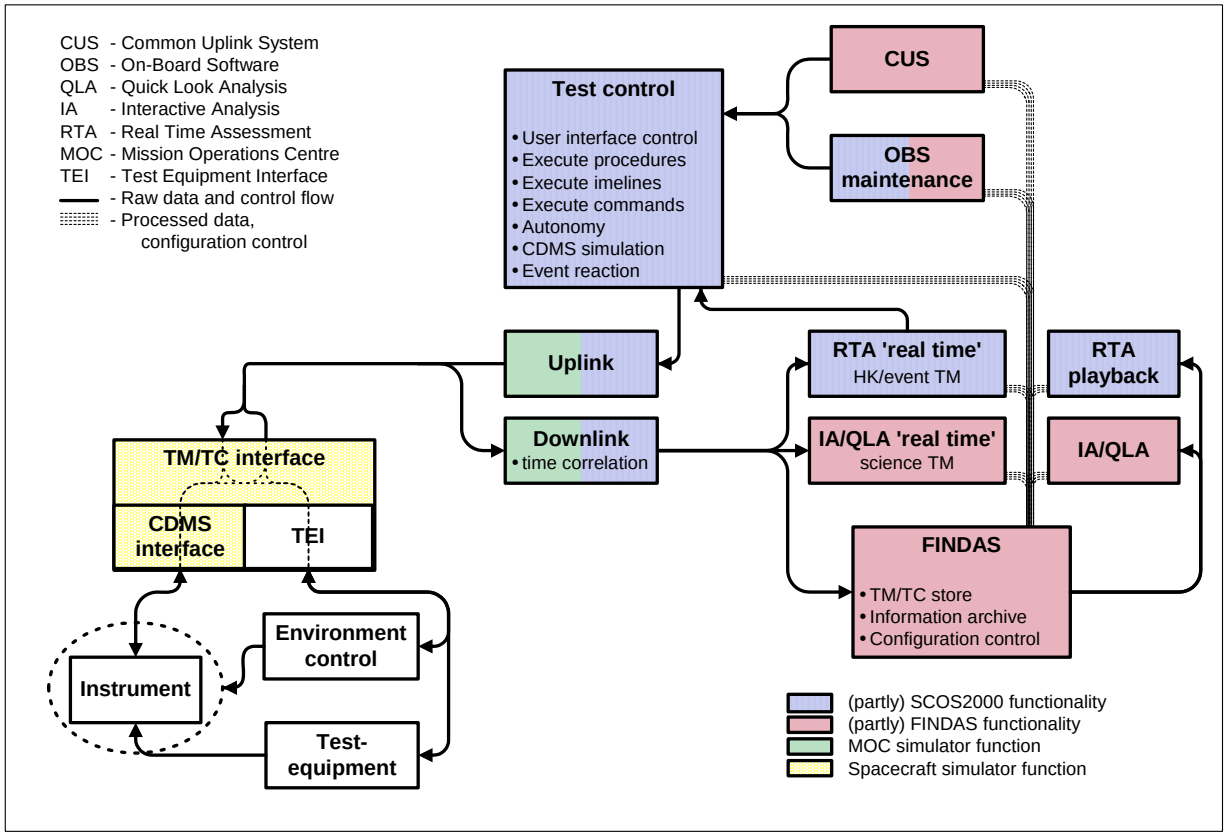

Figure 3-1 Instrument -Level Test Configuration

3.2 System-level Tests

Figure 3-2 shows the EGSE configuration during these tests. The CCE shall provide the same interface to the FINDAS as the MOC operational system (consolidated telemetry).

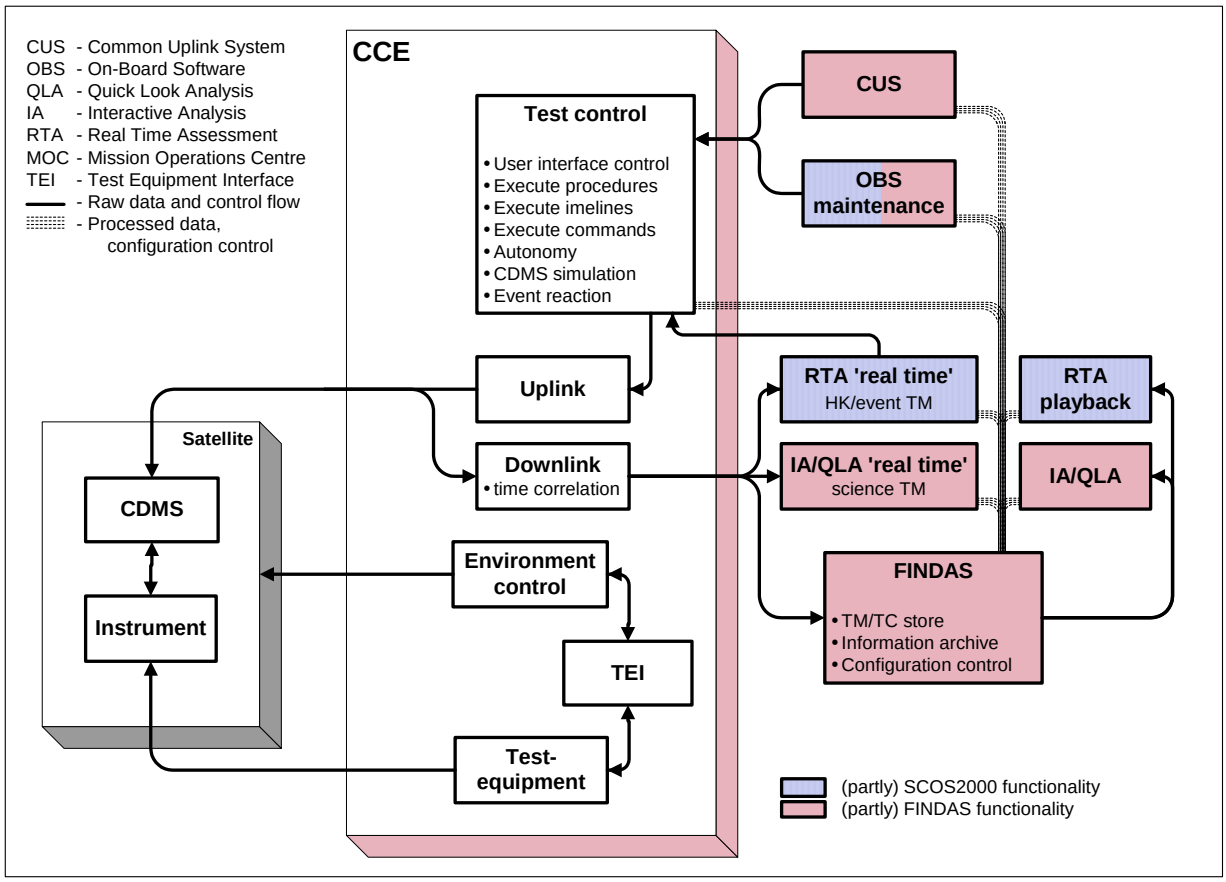

Figure 3-2 System-Level Test Configuration

3.3 Operations

Figure 3-3 shows the EGSE configuration during this time. The EGSE components to the right of FINDAS may be distributed in the Ground Segment (e.g. the CUS at the FSC, the RTA at the ICC and MOC etc).

All control of spacecraft operations will be carried out by the MOC.

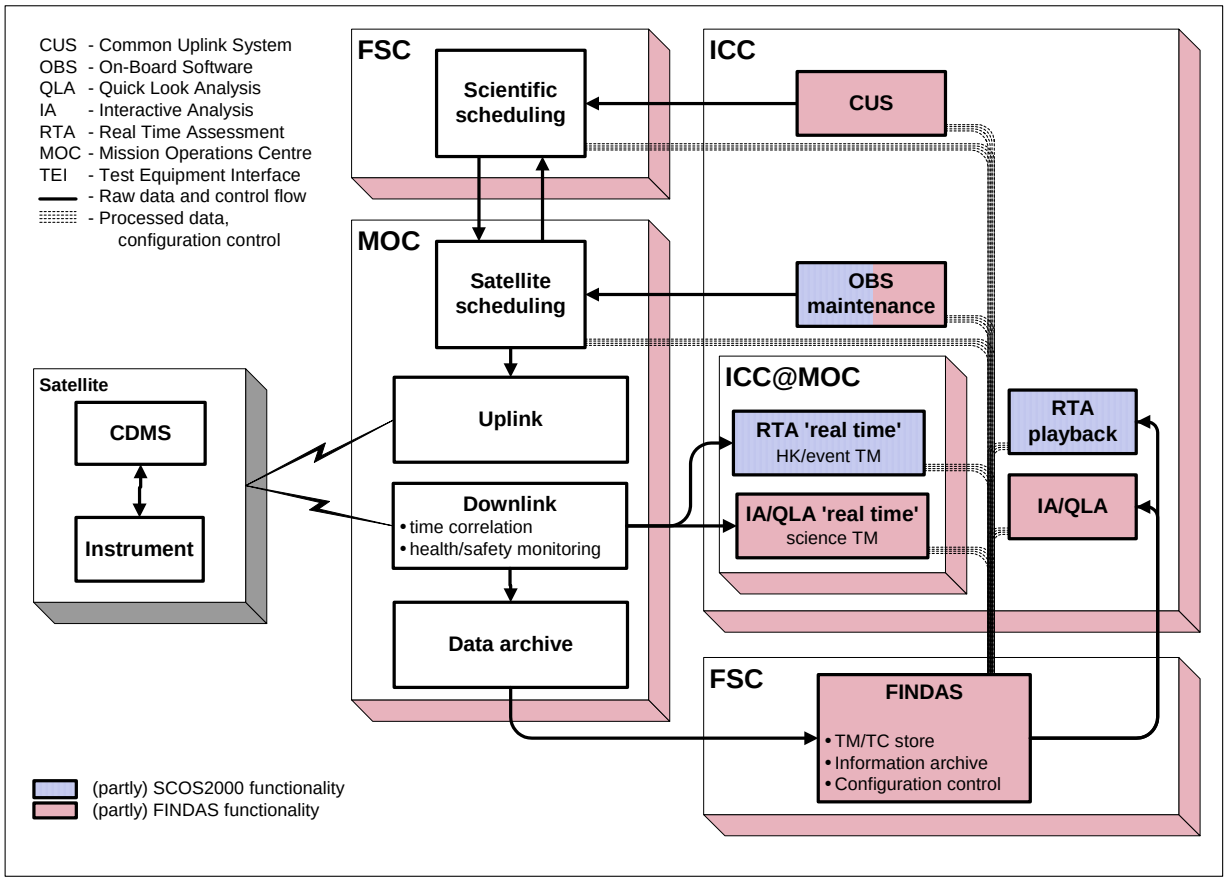

Figure 3-3 Operations Configuration

3.4 Subsystem Tests

Figure 3-4 shows a possible configuration of the EGSE during these tests. This configuration will be used for testing the parts of the instrument warm electronics that interface to the spacecraft (ICU/DPU) before their integration with the rest of the warm electronics.

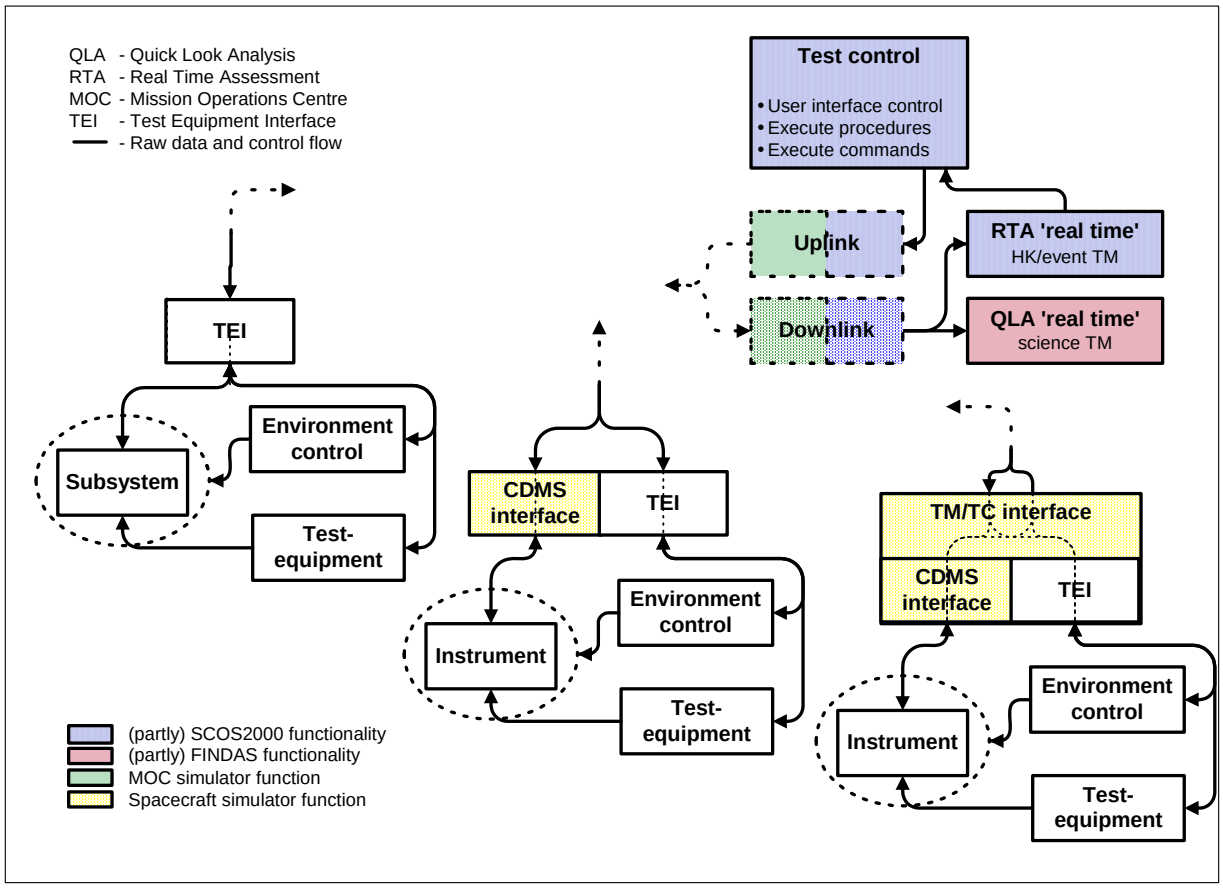

Figure 3-4 Subsystem-Test Configuration

4. EGSE COMPONENT REQUIREMENTS

This section provides the requirements for each component (or group of components) in the EGSE system. Each Requirement is assigned a unique number and an indication of the stage at which the requirement must be fulfilled:

- SUB-D represents a subset of requirements applicable only to early development model subsystem tests.
- SUB indicates the requirement is applicable to the subsystem test
- ILT indicates the requirement is applicable to the instrument-level test environment.
- IST indicates the requirement is applicable to the integrated system test environment.

4.1 Interface Unit

This section includes requirements on the Uplink and Downlink components.

4.1.1 Functional Requirements

IFU-FR-01	SUB-D, SUB, ILT		The interface unit shall contain an interface with the instrument (CDMS,PCS)
IFU-FR-02	SUB-D, SUB, ILT		The interface unit shall contain an unrestricted number of interfaces to test-equipment (TEI-n). *Do we need to provide a list of those test equipment interfaces that should be provided as a minimum?*
IFU-FR-03	*SUB-D, SUB, ILT*		*The interface unit shall accept TC packets from the Test Control component (interface TBD).*
IFU-FR-04	SUB-D, SUB, ILT		The interface unit shall pass TC-packets to the addressed equipment.
IFU-FR-05	SUB-D, SUB, ILT		Each individual Test-Equipment Interfaces (TEI-i) shall be able to unpack a TC-packet and control the corresponding test-equipment accordingly
IFU-FR-06	SUB-D, SUB, ILT		Each individual Test-Equipment Interfaces (TEI-i) shall be able to accept data from the corresponding test-equipment and pack the data in a TM-packet
IFU-FR-07	SUB-D, SUB, ILT		The interface unit shall merge the TM-packets from the TEIs into the TM-stream (*passed to the real-time and TM-store components)*
IFU-FR-08	SUB-D, SUB, ILT		The interface-unit shall pass TM-packets originating from the instrument or from test equipment interfaces *to the real-time and TM-store components.*
IFU-FR-09	SUB-D, SUB, ILT		The interface-unit shall correlate each TM-packet with the ERT *before passing them to the real-time and TM-store components*
IFU-FR-10	SUB-D, SUB, ILT		The interface unit shall be able to handle a TM-data stream from the instrument with a *maximum* data-rate of 400 kbps

IFU-FR-11	*SUB-D, SUB, ILT*		*The interface unit shall be able to handle a TM-data stream from the instrument, plus test equipment, with a maximum data-rate of 400 kbps*
IFU-FR-12	SUB-D, SUB, ILT		The interface unit shall be able to mimic the polling-scenario of the CDMS. *Typical data transfer will take place by transferring one packet at a time in a time slot of 20 ms. 38 out of 50 slots will be available for TM-transfer - see AD1*
IFU-FR-13	SUB-D, SUB, ILT		The interface unit shall be able to send TC-packets at a rate of *at least* 4kbps. *For efficient memory uplink it would be useful to be able to send TC data at a rate 10 times as fast as the nominal uplink rate.*
IFU-FR-14	SUB-D, SUB, ILT		The interface units shall be able to limit the TC-stream to 2 TC-packets per second. *Typical data transfer will take place by transferring one packet at a time in a time slot of 20 ms. 2 out of 50 slots will be available for TC-transfer - see AD1*

4.2 Test Control

4.2.1 Test Procedures

A test procedure in the sense of this document is a list of activities, written in a computer and human readable format. It may contain:

- instrument commands
- commands to the test equipment
- commands to the environment control
- time synchronisation command(s)
- wait instructions for absolute and relative time
- wait instructions for input (from user or from TM via RTA 'real time')
- branches
- loops
- TM read instructions (events and HK via RTA 'real time')
- use of variables and constants
- comments
- integer and floating point calculations
- read/write statements from/into local files or displays (within Test Control)
- string operations
- execution of sub-procedures or other procedures

Note that in SUB and ILT and likely also in IST, the telemetry will contain also dedicated HK packets from test environment and test equipment control. These are therefore also accessible for reading from test procedures via RTA 'real time'.

4.2.2 Command Timelines

During instrument testing, command timelines are the equivalent of the On-board Mission Timeline (MTL, see also FOIRD). Rather than running on the OBDH in flight, they will be executed from Test Control during instrument testing. They are defined as an absolute time tagged sequence of instrument commands. The format of this sequence is still TBD, but it needs to be computer and human readable. All command timelines will be prepared within the Common Uplink System as relative time tagged sequences, absolute times will be assigned to them at runtime when launched by Test Control.

4.2.3 Autonomy Procedure

In order to simulate the autonomy functions of the OBDH within the EGSE, Test Control must be able to run one ore more procedures (at the same time) which can read TM values (events and HK via RTA 'real time'), check them against defined limits and branch according to the result of this comparison. The nominal case is that no actions (commands etc.) are triggered by these procedures. In case an out-of-limit or unexpected event situation is encountered, predefined actions will be triggered. These actions may consist of:

- issue warning message(s) to Test Control or operator
- stop or abort ongoing test procedures or command timelines
- execute predefined sets of commands
- generate log information
- generate events packets

The format and syntax of autonomy procedures may be similar to test procedures. Note that autonomy procedures may run also in between test procedures or command timelines, since although no science TM packets are generated when the instrument is idle or standby, HK packets are still generated.

4.2.4 Requirements

4.2.4.1 Commanding

TC-CMD-01	SUB, ILT, IST		All instrument related commands can be imported from FINDAS.
TC-CMD-02	SUB-D, SUB, ILT, IST		It shall be possible to issue any instrument specific command and manually provide all its required parameters at any time. *Note: Manual commands are always allowed.*
TC-CMD-03	SUB-D, SUB, ILT, IST		All manual commanding activities shall be logged, incl. the time at which every command is sent.
TC-CMD-04	SUB, ILT		Test equipment and environment control commands can be imported from FINDAS.
TC-CMD-05	SUB-D, SUB, ILT		All commands for test equipment and environment control must be accessible from Test Control for immediate execution at any

			time. *Note: Manual commands are always allowed.*
TC-CMD-06	SUB, ILT		It shall be possible to generate and edit test procedures from within Test Control.
TC-CMD-07	SUB, ILT		It shall be possible to import/export test procedures and associated files (ex.: calibration files) from/to FINDAS.
TC-CMD-08	SUB, ILT		It shall be possible to store a TBD number (~10<n<~100) of test procedures within Test Control.
TC-CMD-09	SUB, ILT, IST		It shall be possible to select and execute test procedures from Test Control and provide all user input eventually needed by these procedures.
TC-CMD-10	SUB, ILT, IST		The actual commands generated by the test procedure are logged, incl. the time at which they were sent.
TC-CMD-11	ILT, IST		It shall be possible to import command timelines from FINDAS.
TC-CMD-12	ILT, IST		It shall be possible within few seconds to assign absolute times to relative time tagged command timelines.
TC-CMD-13	ILT, IST		It shall be possible to combine several relative time tagged command timelines into a "schedule" of absolute time tagged command timelines.
TC-CMD-14	ILT, IST		It shall be possible to execute an absolute time tagged command timeline or schedule. The individual commands are launched automatically at the exact time they are due.
TC-CMD-15	ILT, IST		A set of a TBD number (~10<n<~100) of relative time tagged command timelines can be stored within Test Control to be available for immediate execution (implies fast assignment of absolute time).
TC-CMD-16	SUB, ILT, IST		It shall be possible to export all logging information from manual commanding and test procedures to FINDAS.
TC-CMD-17	SUB-D, SUB, ILT		It shall be possible to add to and maintain an electronic log of the ongoing test activities.
TC-CMD-18	SUB, ILT, IST		It shall be possible to import any required on-board software together with the related commands for its uplink, as prepared by OBS Maintenance, from FINDAS.

TC-CMD-19	SUB, ILT, IST		It shall be possible to abort, stop and resume any test procedure and command timeline by manual interaction.
TC-CMD-20	SUB, ILT		It shall be possible to synchronise the whole EGSE setup (simulates spacecraft reference time updating).
TC-CMD-21	SUB-D, SUB, ILT, IST		The syntax and parameter range of manual commands, commands from test procedures and commands from timelines/schedules will be checked automatically within Test Control before they are sent to the instrument.
TC-CMD-22	SUB, ILT, IST		In case of erroneous commands in the sense of 2.21. a message will be displayed to the Test Control user and current procedures or timelines will be stopped or suspended.
TC-CMD-23	SUB, ILT, IST		It is highly desirable that test procedures and command timelines are written in the same language or generated using the same language.

4.2.4.2 Autonomy Functions

TC-AUT-01	ILT		It shall be possible to generate and edit autonomy procedures from within Test Control.
TC-AUT-02	ILT		It shall be possible to import/export autonomy procedures and associated files from/to FINDAS.
TC-AUT-03	ILT		It shall be possible to run one or more autonomy procedures in parallel during a test procedure or a command timeline from within Test Control.
TC-AUT-04	ILT		It shall be possible to start, abort, stop and resume any autonomy procedure by manual interaction from Test Control.
TC-AUT-05	ILT		Autonomy procedures shall be able to read TM values (events + HK)
TC-AUT-06	ILT		It is highly desirable that autonomy procedures have same syntax as test procedures and are written in the same language.
TC-AUT-07	ILT		It shall be possible to run autonomy procedures also during instrument idle or standby modes.
TC-AUT-08	ILT		Autonomy procedures may generate event packets
TC-AUT-09	ILT		Autonomy procedures may generate log

			information
TC-AUT-10	ILT		Autonomy procedures may stop or abort ongoing test procedures or command timelines
TC-AUT-11	ILT		Autonomy procedures may generate warning messages to the operator

4.2.4.3 User Interface

TC-UI-01	SUB-D, SUB, ILT		All Test Control specific activities are carried out via computer keyboard(s) and/or mouse (mice).
TC-UI-02	SUB, ILT		It shall be possible to track an ongoing test procedure via a display indicating the position within the procedure and showing at least the recent past and near future steps of the procedure. Back and forward scrolling shall be possible.
TC-UI-03	ILT, IST		It shall be possible to track a command timeline via a display indicating the position within the timeline and showing at least the recent past and near future of the command timeline. Back and forward scrolling shall be possible.
TC-UI-04	SUB-D, SUB, ILT, IST		At any moment the manual command window shall be accessible.
TC-UI-05	SUB, ILT		At any moment the message window for the feedback from autonomy functions and test procedures shall be displayed.
TC-UI-06	SUB-D, SUB, ILT		On user request it shall be possible to display the logging information originating from manual commanding or test procedures.
TC-UI-07	SUB-D, SUB, ILT, IST		There shall be a separate window from where test procedures, autonomy procedures or command timelines/schedules can be selected, started, aborted, stopped and resumed.
TC-UI-08	SUB-D, SUB, ILT, IST		Both, the spacecraft reference time (= simulated test time ≠ local time) at 1 sec resolution and the actual local time are displayed to the Test Control user.
TC-UI-09	SUB-D, SUB, ILT, IST		Manual commanding shall be supported by tools which allow sending one single command: at a specified absolute time, after a specified relative time or for immediate execution. Commands submitted for execution at or after a specified time shall be removable while not yet executed.

TC-UI-10	SUB, ILT		There shall be a separate window in which editing of autonomy and test procedures and logs is possible. Export and import activities from/to Test Control shall be carried out here as well.
TC-UI-11	SUB, ILT		There shall be a separate window in which export and import activities from/to Test Control are carried out.

4.2.4.4 General EGSE requirements

TC-GEN-01	SUB, ILT		Real time voice communication between Test Control, Test Environment control and Test Equipment control shall be possible.
TC-GEN-02	SUB, ILT		The user of Test Control shall see and shall be able to select the RTA ('real time') display(s) of interest.
TC-GEN-03	ILT		The user of Test Control shall see and shall be able to select the IA/QLA ('real time') display(s) of interest.

4.3 FINDAS

TBW

4.4 RTA

These requirements have been taken from the FIRST /Planck RTA URD (RD2) plus additional requirements (RD7).

Following the issue of this document SCOS2000 has been evaluated as a potential RTA implementation and the results of this evaluation (RD6) have been used to update the requirements. Requirements not implemented by the current SCOS2000 version are noted in blue.

4.4.1 General Description

4.4.1.1 General capabilities

4.4.1.2 General constraints

The RTA system will operate within the FINDAS environment. TM packets and uplinked command information must be fetched from FINDAS.
Whenever this document mentions the term file it might be a FINDAS object as well. Later versions of the URD must specify the real source or destination for RTA.

4.4.1.3 User characteristics

The RTA system will be operated by trained people. In this document the term operator will be used to specify the user of the RTA system.

4.4.1.4 Operational environment

FINDAS will be responsible for the TM data archive, providing access to both on-line TM data and archived TM data. Instrument on-board memory dumps will also be stored within FINDAS.

4.4.2 Functional Requirements

4.4.2.1 Instrument status displays

RTA-DISP-01	SUB, ILT, IST	Essential	RTA shall be able to display any TM packet.
RTA-DISP-02	SUB, ILT, IST	Essential	RTA shall be able to display HK parameters in alpha-numeric displays. *The alpha-numeric displays shall have a two-dimensional area where fixed text and the current HK parameter values can be laid-out.*
RTA-DISP-03	SUB, ILT, IST	Essential	Alphanumeric displays shall be able to display parameters in raw or converted (i.e. engineering) units.
RTA-DISP-04	SUB, ILT, IST	Essential	In alphanumeric displays, parameter values shall be displayed in colour indicating their status (e.g. red if out of limits, green if within limits).
RTA-DISP-05	SUB, ILT, IST	Essential	It shall be possible to define alphanumeric displays with a free layout. This layout shall consist of the following types of fields: • *fixed (non-variable) text* • *updated fields showing the actual value of TM parameters* *These fields can be placed anywhere in a given frame representing the alphanumeric display.*
RTA-DISP-06	SUB, ILT, IST	Desireable	RTA shall be able to display HK parameters in extended alpha-numeric displays (AND) including bar-graphs, dials... Not implemented in SCOS2000 *Note: Mimic displays can be an (expensive) way to create an extended AND*
RTA-DISP-07	SUB, ILT, IST	Desirable	It shall be possible to configure to edit the layout of an (extended) alpha-numeric display while RTA is operating. Not implemented in SCOS2000 *It is only possible to copy-paste a parameter on to a display line. The format of the parameter value – raw or converted, binary, octal, decimal, hexadecimal – can be modified.*
RTA-DISP-08	SUB, ILT, IST	Desirable	It shall be possible to save the layout of an (extended) alpha-numeric display. Not implemented in SCOS2000
RTA-DISP-09	SUB, ILT, IST	Desirable	It shall be possible to restore the layout of an (extended) alpha-numeric display. Not implemented in SCOS2000

RTA-DISP-10	SUB, ILT, IST	Essential	It shall be possible to generate a tabular listing of one or more instrument parameters showing the history of them.
RTA-DISP-11	SUB, ILT, IST	Essential	The SCOS 2000 listing display shall work in replay mode as well as in real-time mode.
RTA-DISP-12	SUB, ILT, IST	Essential	The SCOS 2000 listing display shall provide a scrolling window.
RTA-DISP-13	SUB, ILT, IST	Essential	The number of lines saved in the scrolling window shall be configurable.
RTA-DISP-14	SUB, ILT, IST	Essential	It shall be possible to generate stripchart like plots of one or more instrument parameters showing the history of them.
RTA-DISP-15	SUB, ILT, IST	Essential	It shall be possible to select, at any time, the time period for which a listing or stripchart shows data. Listings: not implemented in SCOS2000 Stripcharts: implemented in SCOS2000 *Note: Selecting the retrieval time of any view in the TM monitor desktop affects all the other views.*
RTA-DISP-16	SUB, ILT, IST	Essential	It shall be possible to display several different instrument status display (including different ones of the same type) at the same time.
RTA-DISP-17	SUB, ILT, IST	Essential	It shall be possible to generate hard-copies of all instrument status displays.
RTA-DISP-18	SUB, ILT, IST	Desirable	Synoptic (mimic) displays shall give an overview about the instrument setup
RTA-DISP-19	SUB, ILT, IST	Desirable	Synoptic displays shall also be able to display alphanumeric values.
RTA-DISP-20	SUB, ILT, IST	Desirable	It shall be possible to generate a raw dump of a TM packet or part of it.

4.4.2.2 Anomaly monitor

RTA-MON-01	SUB, ILT, IST	Essential	All incoming TM packets must be monitored to capture the following events: • out-of-limits • unexpected status • conversion errors • TM stream drop • TM transmission errors
RTA-MON-02	SUB, ILT, IST	Essential	The anomaly events shall be tagged with the on-board time of the anomaly occurrence and with the RTA system time.
RTA-MON-03	SUB, ILT, IST	Essential	There shall be a log like displays to show the anomaly events.
RTA-MON-04	SUB, ILT, IST	Essential	There shall be a display, which shows all current anomaly events.
RTA-MON-05	SUB, ILT, IST	Essential	The display of anomaly events shall always display the actual value of the TM parameter.

			In the SCOS 2000 OOL display, which shows all current out-of-limit parameters the column "value" is misleading because it shows the value of the parameter at the time it changed its state to out-of-limit.
RTA-MON-06	SUB, ILT, IST	Essential	The operator should be able to select or filter the type of anomaly events to be displayed or logged in a file.
RTA-MON-07	SUB, ILT, IST	Essential	It shall be possible to store all anomaly events reports.
RTA-MON-08	SUB, ILT, IST	Essential	RTA must identify "Event Packets" from the instrument, monitor and log them and flag to the operator any of high criticality.
RTA-MON-09	SUB, ILT, IST	Essential	It shall be possible to generate OOL event packets from the behaviour checker in replay mode.

4.4.2.3 Command verification

Because the FIRST instruments will be operated most of the time without real time contact all instrument commands must be verified on-board. Either for all or only for the failed commands an "info" TM packet will be generated.

RTA-CVER-01	SUB, ILT, IST	Essential	RTA must identify "command verification" TM packets from the instrument, monitor and log them and flag to the operator any of high criticality.
RTA-CVER-02	SUB, ILT, IST	Essential	The RTA shall verify the command reception at the instrument.
RTA-CVER-03	SUB, ILT, IST	Essential	Each uplinked instrument command shall be checked for corresponding HK parameter change

4.4.3 Performance requirements

RTA-PERF-01	SUB, ILT, IST	Essential	All RTA modules must be able to process the TM data in real-time. To get a safety margin the average system load during real-time processing shall not be greater then 50% of available resources.
RTA-PERF-02	SUB, ILT, IST	Essential	There may be a temporarily delay of TBD seconds during real-time processing which must be caught up in TBD seconds.
RTA-PERF-03	SUB, ILT, IST	Essential	The real-time performance shall not be corrupted by other processes running on the system
RTA-PERF-04	SUB, ILT, IST	Essential	RTA processes shall be compartimentalised such that operator (input) errors in one process have negligible influence on any other process.
RTA-PERF-05	SUB, ILT, IST	Desirable	The time needed to start up the RTA system shall not exceed 10 minutes, not counting the time

			needed to boot the operating system.
RTA-PERF-06	SUB, ILT, IST	Essential	It shall be possible to upload a new MIB within 10 minutes. *This shall include the time necessary to trigger or to stop and start processes of the system.*

4.4.4 Interface requirements

RTA-IF-01	SUB, ILT, IST	Essential	The RTA system shall fetch the TM packets on-line from FINDAS.
RTA-IF-02	SUB, ILT, IST	Essential	The RTA system shall fetch archived TM packets from FINDAS.
RTA-IF-03	SUB, ILT, IST	Essential	The RTA system shall fetch information about uplinked commands from FINDAS.
RTA-IF-04	SUB, ILT, IST	Essential	The RTA system shall be able to read from objects in FINDAS. Not implemented in SCOS2000 *SCOS2000 uses ASCII "tabular" files to read in the MIB data.* *There are interfaces available within SCOS2000 to allow add-ons which could communicate with FINDAS*
RTA-IF-05	SUB, ILT, IST	Essential	The RTA system shall be able to create/update objects in FINDAS. Not implemented in SCOS2000 *There are interfaces available within SCOS2000 to allow add-ons which could communicate with FINDAS*

4.4.5 Operational requirements

RTA-OPS-01	SUB, ILT, IST	Essential	All RTA modules must handle all operator input errors in a save and consistent way. No operator input error shall stop the correct RTA processing.
RTA-OPS-02	SUB, ILT, IST	Essential	Operator input shall be logged (e.g. command input, on-line changes to MIB data).
RTA-OPS-03	SUB, ILT, IST	Essential	Operator (input) errors shall never corrupt the original telemetry
RTA-OPS-04	SUB, ILT, IST	Essential	RTA shall be able to run in a "default" mode i.e. without operator's input. In this case RTA is driven by AOT ID or any suitable identifier.
RTA-OPS-05	SUB, ILT, IST	Essential	There shall be no need to configure manually the RTA for different TM packet types
RTA-OPS-06	SUB, ILT, IST	Essential	It shall be possible to replay archived TM data.
RTA-OPS-07	SUB, ILT, IST	Essential	It shall be possible to select different playback speeds, for example, twice real time or just 1/4 of real time.

RTA-OPS-08	SUB, ILT, IST	Essential	It shall be possible to run more than one RTA "Sessions". *If there is one RTA session to check on-line data it shall be possible to replay archived TM data without the need for switching machines.*
RTA-OPS-09	SUB, ILT, IST	Essential	The standard (default) RTA configuration shall be controlled by files.
RTA-OPS-10	SUB, ILT, IST	Essential	The RTA startup in default configuration shall be simple. *I.e. it shall be controlled by a configuration file or script*
RTA-OPS-11	SUB, ILT, IST	Essential	The operator shall be given the option to go back to the standard RTA configuration at any time.
RTA-OPS-12	SUB, ILT, IST	Essential	The operator shall be able to select any combination of instrument status displays, anomaly monitors and command verification tools.
RTA-OPS-13	SUB, ILT, IST	Essential	There shall be a display available showing the current RTA setup
RTA-OPS-14	SUB, ILT, IST	Essential	The definition of TM packets and their contents shall be stored in files which are loaded during start-up of RTA or later on operator request.
RTA-OPS-15	SUB, ILT, IST	Essential	The instrument parameter definitions shall allow the specification of: • derived parameters • special conversion functions • special out-of-limit checks
RTA-OPS-16	SUB, ILT, IST	Desirable	It shall be possible to update the TM packet definition used by RTA in on-line mode. It shall be possible to store the new definition into a file. Not implemented in SCOS2000
RTA-OPS-17	SUB, ILT, IST	Essential	The definition of instrument command characteristic used by command verification shall be stored in files which are loaded during start-up of RTA or later on operator request
RTA-OPS-18	SUB, ILT, IST	Desirable	It shall be possible to update the instrument command definitions used by RTA in on-line mode. It shall be possible to store the new definition into a file. Not implemented in SCOS2000
RTA-OPS-19	SUB, ILT, IST	Essential	It shall be possible to define a set of displays for the operator's desktop.
RTA-OPS-20	SUB, ILT, IST	Essential	The operator shall have the possibility to select one of these display sets like he/she selects a single display.
RTA-OPS-21	SUB, ILT, IST	Essential	It shall be possible that the set of displays shown by the operator's desktop is selected by the actual value of TM parameter

4.4.6 Man-machine interface

RTA-MMI-01	SUB, ILT, IST	Essential	There shall be a hard-copy generation facility for all displays.

4.5 QLA/IA

In all configurations used for instrument testing the QLA/IA has to interface to the Downlink component for 'real-time' telemetry input and to FINDAS for extraction of telemetry, access to configuration tables and parameters used for processing the incoming data and for storage of test data results (products and logs).

4.5.1 Interface Description

The interface can be divided into three parts, dealing with real-time telemetry, playback telemetry and other data. These interfaces may be implemented as completely different hardware/software components.

4.5.1.1 Real Time Telemetry

During operations the data received from the Satellite is archived in the Data Distribution System (DDS) of the MOC, as it is received, in the form of telemetry source packets. (The Downlink components of the EGSE and the CCE are responsible for simulating this process during ground testing.) The DDS does not distribute this data, but makes it available to other systems on request. Data may be requested by packet type (events, housekeeping, science), APID and/or time period.

Real time data is requested from the DDS on a packet by packet basis by QLA/IA. The protocol for this is TBD, but we may assume something like a client-server interaction using TCP/IP sockets.

The hardware interface to the DDS is still TBD, but we assume a standard Ethernet link.

4.5.1.2 Playback Telemetry

When all the data for a given time period (TBD) is available, the data is ingested, by an FSC process, into FINDAS and made available to the rest of the ground segment. From here it may be requested in many ways according to the object model employed in the FINDAS database. It is this interface, which is used for playback of data into QLA/IA (note; the use of the term 'playback' is misleading as the data is requested from FINDAS, as required, rather than sent from FINDAS to be dealt with by QLA/IA as it arrives).

The interface to FINDAS is still TBD, but we assume a client-server protocol over a network link.

4.5.1.3 Other Data

Other data is read/written from/to FINDAS using the standard network interface.

4.5.2 Requirements

4.5.2.1 Packet Handling

These requirements deal with the telemetry packet handling as seen by the QLA/IA software. They should be met both during real-time and playback operation.

QLAIF - PH-01	ILT, IST		The I/F shall accept all types of telemetry source

			packets *The different types will be defined in the Packet Structure ICD (RD2).*
QLAIF - PH-02	ILT, IST		It shall be possible to select types of telemetry source packet. Selection shall be by APID and/or packet type and subtype. *Selection of packets will reduce the performance overhead of dealing with unnecessary packets such as, for example, event packets, which are not normally handled by QLA/IA*
QLAIF - PH-03	ILT, IST		The I/F shall convert telemetry source packets into data structures compatible with the analysis software environment used by QLA/IA. *For example, the interface should create an IDL data structure from each packet that can them be accessed by the QLA/IA software without knowledge of the form in which the data arrived at the interface.* *No conversion of data values into engineering units will be provided - this is a function of the QLA/IA software.*
QLAIF-PH-04	ILT, IST		The interface shall allow multiple processes to make requests at the same time. *QLA/IA may run as several processes operating in parallel.*
QLAIF-PH-05	ILT, IST		The interface shall allow a request for telemetry packets over a given absolute time range. *This will allow QLA to generate products related to a given period.*

4.5.2.2 Real Time

QLAIF-RT-01	ILT, IST		The I/F shall be able to handle telemetry source packets at the maximum rate they are generated by the instrument and additional test equipment *The instrument may generate data at a rate exceeding the average on-board data rate (100 kbps, TBC) in bursts, during testing. The additional test equipment may also generate data at a high rate. The interface should be able to transfer data at this maximum rate to prevent the QLA running behind the real-time telemetry*
QLAIF-RT-02	ILT, IST		The interface shall allow a request for the last telemetry packet, of a given type, to become available.
QLAIF-RT-03	ILT, IST		The interface shall allow a request for the next telemetry packet, of a given type, to become available. *In this way the QLA software may continue to use*

			the latest available telemetry data.
QLAIF-RT-04	ILT, IST		The interface shall allow a request for telemetry packets over a given time range relative to the current time. *This will allow QLA to 'look back' over a time range to generate products.*
QLAIF-RT-05	ILT, IST		The interface shall conform to the DDS hardware interface
QLAIF-RT-06	ILT, IST		The interface shall conform to the DDS access protocol interface

4.5.2.3 Playback

QLAIF-PB-01	ILT, IST		The I/F shall be able to request telemetry source packets based on a query of the FINDAS database *QLA/IA will need to access data in many ways, based on the object model defined for the FINDAS database.*
QLAIF-PB-02	ILT, IST		The interface shall conform to the FINDAS hardware interface
QLAIF-PB-03	ILT, IST		The interface shall conform to the FINDAS access protocol interface

4.5.2.4 Other Data

QLAIF-OD-01	ILT, IST		The I/F shall be able to request any data resident in the FINDAS database based on a query. *QLA/IA will need to access other data (e.g. calibration files, parameter data, spacecraft data etc .in many ways, based on the object model defined for the FINDAS database.*
QLAIF-OD-02	ILT, IST		The I/F shall allow writing of data to the FINDAS database, when authorised. *The interface will not check the authorisation. It will provide the relevant information to FINDAS to authorise the transaction.*
QLAIF-OD-03	ILT, IST		The interface shall conform to the FINDAS hardware interface
QLAIF-OD-04	ILT, IST		The interface shall conform to the FINDAS access protocol interface

4.6 CUS

4.6.1 General Description

The Common Uplink System is a tool for generating relative time tagged command sequences for all FIRST instrument measurement types: general astronomical observations, calibration observations and engineering measurements. The CUS itself has three functions:

a) to translate the definition of a measurement (provided as a 'script') into the corresponding command sequence(s).

b) to provide an environment for the editing and development of 'scripts'
c) to convert astronomical observations (in the form of filled-out Astronomical Observation Templates (AOTs) into measurement 'scripts'.

(The FIRST Common Uplink System will be based on the ISO experience (cf. Baseline Requirements Document for the ISO Calibration Uplink System, (RD11)). A discussion of the applicability of the ISO system to the above measurement types has been presented in some detail in the technical note "Towards a Common Uplink System for FIRST" (RD12)).

The responsibility for the development and maintenance of the CUS lies with the ICCs. The split in responsibility between them is TBD.

4.6.2 Interfaces

The outcome of the CUS tool, relative time tagged command sequences, are meant as input to the FSC/MOC mission planning tools, or in the ILT case for Test Control.

Note: At the front end there are some interfaces to data bases and "helpers" (e.g. AOT online help, instrument documentation etc.), which are not considered as part of the CUS and should
be discussed elsewhere. Also the AOT User Interface which will have to be developed by FSC is not considered here, however the underlying AOT logic driving this interface will be part of the CUS.

4.6.3 Users

Users of the CUS will be:

a) the instrument experts at the ICCs, for generation of calibration observations and engineering measurements. (AOT type observations will be handled as for every general observer by using the AOT User Interface).
b) Cross-calibration scientists, for the generation of cross-calibration observations
c) General astronomers using the satellite.

4.6.4 Capability Requirements

CUS-CAP-01	ILT, IST	Essential	The CUS shall offer a flexible editing and development environment for writing measurement 'scripts'
CUS-CAP-02	ILT, IST	Essential	It shall be possible to construct an all types of measurement using a 'script'.
CUS-CAP-03	ILT, IST	Essential	The 'script' shall provide the following capabilities: • start and end statements to explicitly bracket a measurement entity • spacecraft related pointing commands (incl. all scientific pointing modes offered by FIRST) • single instrument commands (incl. dummy

			commands) with their respective parameters represented by "verb elements" of the scripting language • all onboard control procedures (OBCP) as stored onboard with their respective parameters represented by "verb elements" of the scripting language • default parameter settings shall be identified by a special character • flags to disable certain parameter and limit checks in the subsequent translation process needed for special engineering procedures • branches • loops • use of variables and constants • integer and floating point calculations • string operations • comments • statements to write to displays • I/O statements to allow the input/output of information (configuration data, logs etc) • call and/or inclusion of sub-routines, scripts or other routines with and without parameters to allow easy access to already written batch code • call of functional units (the meaning of functional units follows the abstraction level description in the PACS ICC software URD, note that a functional unit may consist of a combination of OBCPs and single instrument commands
CUS-CAP-04	ILT, IST	Essential	It shall be possible to modify or adapt already existing and stored 'scripts'.
CUSCAP-05	ILT, IST	Essential	It shall be possible to translate a 'script' into a relative time-tagged command sequence. *The format is TBD, but will be compatible with that required by the FSC as input for the mission planning system and command line generator or, for ILT cases, that required by Test Control.*
CUS-CAP-06	ILT, IST	Essential	It shall be possible to start the translator from the 'script' editing environment.

CUS-CAP-07	ILT, IST	Essential	The translator shall check the 'script' for correctness of syntax
CUS-CAP-08	ILT, IST	Essential	The translator shall check whether the provided parameters for the individual commands have valid values and are within allowed limits
CUS-CAP-09	ILT, IST	Essential	It shall be possible to disable the parameter checking for certain blocks or individual statements.
CUS-CAP-10	ILT, IST	Essential	An error message log shall be provided during all translation runs.
CUS-CAP-11	ILT, IST	Essential	In case of errors it shall be possible to switch from the translator to the editing environment. *Ideally the environment should start up at the location of the first error*
CUS-CAP-12	ILT, IST	Essential	It shall be possible to store the generated command sequences (in FINDAS)
CUS-CAP-13	ILT, IST	Essential	During the translation process a detailed report shall be generated and stored, which provides TBD information about functional units. (e.g. the total duration of the observation/measurement or detailed durations of its individual building blocks or functional units.)
CUS-CAP-14	ILT, IST	Essential	The system shall at any time be able provide the total duration of the measurement as well as the duration of individual functional units. *This is required for feedback to the AOT logic which is communicating with the user. It forms the basis for an Observation Time Calculator*
CUS-CAP-15	ILT, IST	Essential	It shall be possible to convert astronomical observations (in the form of filled-out Astronomical Observation Templates (AOTs) into measurement 'scripts'
CUS-CAP-16	ILT, IST	Essential	A User Interface shall be provided that has has the following properties: • It shall pass all required parameters to the CUS • It shall allow the display of all messages and reports from the CUS • It shall allow the translation process to be started • It shall be possible to manage (copy, rename,

			delete, etc.) the stored results of the translation process.

4.7 OBS Maintenance

4.7.1 General Description:

On-board software is not only needed for the DPU/ICU but also for other microprocessor-powered instrument subunits (e.g. SPU). In the following requirements, though only the DPU/ICU is mentioned, they also apply to those subunits.

The RAM images shall be stored within FINDAS because of following reasons:

- availability for all applications which need it
- access control by FINDAS
- version control by FINDAS

4.7.2 Requirements:

4.7.2.1 Capability requirements:

OBS-CAP-01	SUB, ILT, IST	Essential	The on-board S/W maintenance environment shall provide the framework to generate and test RAM images and RAM patches (TBC) for the DPU/ICU.
OBS-CAP-02	SUB, ILT, IST	Essential	The on-board S/W maintenance environment shall be able to upgrade via uplink all the modules of the DPU/ICU on-board S/W. Therefore an interface to the uplink system is needed.
OBS-CAP-03	SUB, ILT, IST	Essential	The on-board S/W maintenance environment shall be able to compare downlinked memory dumps with reference memory dumps.
OBS-CAP-04	SUB, ILT, IST	Essential	The on-board S/W maintenance environment shall provide the framework to test on-board data compression algorithms.

4.7.2.2 Constraint requirements:

OBS-CON-01	SUB, ILT, IST	Essential	The on-board S/W maintenance facility shall store RAM images within FINDAS.
OBS-CON-02	SUB, ILT, IST	Essential	The on-board S/W maintenance facility shall fetch RAM images from FINDAS.

4.8 Additional Tools

4.8.1 MIB Editor:

4.8.1.1 General Description:

The MIB contains following data:

- characteristics of the MIB itself
  - description
  - version
  - history
- definition of the downlink data
  - packet structure
  - extraction data for TM parameters
  - conversion tables
  - out-of-limit checks
  - display definitions
- definition of the uplink data
  - packet headers
  - command definitions
  - command sequence definitions
  - command verification
  - command parameter calibrations
  - command parameter checks

The MIB is mainly used by SCOS 2000, but IA(QLA) will also access the MIB. The MIB shall be stored within FINDAS because of following reasons:

- availability for all applications which need it
- access control by FINDAS
- version control by FINDAS

4.8.1.2 Requirements:

4.8.1.2.1 Capability Requirements:

OBS-CAP-01	SUB, ILT, IST	Essential	The MIB editor shall display forms to enter the data.
OBS-CAP-02	SUB, ILT, IST	Essential	Each form shall display all relevant fields for one MIB item, e.g.: • TM packet (fixed, variable) • TM parameter (raw TM parameter, synthetic parameter, constants) • TM parameter calibration (numerical, textual, polynomial) • TM parameter checks • TM parameter display • command packet headers • command definitions • command sequence definitions

			• command verification • command parameter calibrations (numerical, textual) • command parameter checks
OBS-CAP-03	SUB, ILT, IST	Essential	Each form shall have links to other relevant MIB items, eg. the TM parameter form shall have links to the TM packets which contain this parameter, to the corresponding TM parameter calibration form, ...
OBS-CAP-04	SUB, ILT, IST	Essential	If available the MIB editor shall enter default values in an empty form.
OBS-CAP-05	SUB, ILT, IST	Essential	The form shall be checked against empty but mandatory fields.
OBS-CAP-06	SUB, ILT, IST	Essential	All user input shall be checked against: • the allowed range • inconsistencies with other fields
OBS-CAP-07	SUB, ILT, IST	Essential	Whenever an MIB item will be deleted by the user the MIB editor shall warn the user this operation results in orphaned MIB items. The user shall have the option to delete this orphaned MIB items
OBS-CAP-08	SUB, ILT, IST	Essential	For every input field which allows only a limited set of values the MIB editor shall display a menu of this set to select from.
OBS-CAP-09	SUB, ILT, IST	Essential	If the user modifies a field which changes the interpretation of other fields those fields shall be cleared and, if necessary, the list of allowed values shall be updated.
OBS-CAP-10	SUB, ILT, IST	Essential	It shall be possible to make hard copies of the different forms.
OBS-CAP-11	SUB, ILT, IST	Essential	It shall be possible to print summaries.
OBS-CAP-12	SUB, ILT, IST	Essential	The MIB editor help system shall provide help texts for: • each form • each data field in a form (tips on a field)
OBS-CAP-13	SUB, ILT, IST	Essential	It shall be possible to merge to MIBs.
OBS-CAP-14	SUB, ILT, IST	Essential	It shall be possible to perform consistency checks on a loaded MIB
OBS-CAP-15	SUB, ILT, IST	Essential	A history of all major edits shall be recorded.

4.8.1.2.2 Constraint Requirements:

OBS-CON-01	SUB, ILT, IST	Essential	The MIB shall be stored in FINDAS.
OBS-CON-02	SUB, ILT, IST	Essential	The MIB shall be loaded from FINDAS.
OBS-CON-03	SUB, ILT, IST	Essential	The MIB shall contain all data fields specified in the SCOS 2000 MIB Import ICD (RD8).
OBS-CON-04	SUB, ILT, IST	Essential	It shall be possible to import SCOS 2000 MIB data

			files.
OBS-CON-05	SUB, ILT, IST	Essential	It shall be possible to merge SCOS 2000 MIB data files to an existing MIB.
OBS-CON-06	SUB, ILT, IST	Essential	It shall be possible to export SCOS 2000 MIB data files.