SOFTWARE USERS MANUAL

FOR THE

LRO-LAMP INSTRUMENT

January 2009

Draft 5

SwRI® Project 11239

Document No. 11239-LAMP_SUM-01

Contract NNG05EC87C

Prepared by

SOUTHWEST RESEARCH INSTITUTE®
Automation and Data Systems Division
6220 Culebra Road, San Antonio, Texas 78228-0510
(210) 684-5111 • FAX (210) 522-5499

SOFTWARE USERS MANUAL
FOR THE
LRO-LAMP INSTRUMENT

SwRI Project 11239

Document No. 11239-LAMP_SUM-01

Contract NNG05EC87C

Prepared by: Maarten Versteeg
SwRI Software Project Manager

Approved by: Robert Klar
SwRI Section Manager

Approved by: /s/ Ron Black
SwRI Project Manager

Approved by: /s/ P. Brian Gupta
SwRI Quality Assurance

Approved by: /s/ L. D. McCullough
SwRI Project CM

Signatures on file Document Control.

Automation and Data Systems Division
Southwest Research Institute
P. O. Drawer 28510
6220 Culebra Road
San Antonio, Texas 78228-0510
(210) 684-5111

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

REVISION NOTICE

Status	Rev./Chg.	Date	Notes
Draft 0	Rev 0 Chg 0	March 2006	Initial draft started from P-Alice baseline. Includes appendix on specific mini-LIIS operations
Draft 1	Rev 0 Chg 1	September 2006	
Draft 2	Rev 0 Chg 2	April 2007	Updated with LAMP FM Instrument
Draft 3	Rev 0 Chg 3	February 2008	Updated with latest LAMP FM results from the environmental tests, around delivery of LAMP FM to Goddard Space Flight Center for integration on LRO
Draft 4	Rev 0 Chg 4	July 2008	Few TM field errors corrected, parameter file conversion typo corrected and latest set of error codes ingested.
Draft 5	Rev 0 Chg 5	January 2009	Extended LTS control description, corrected some spelling errors

1. SCOPE

1.1 Project Identification

Project Title:	LRO-LAMP: Flight and Ground Software
Project Number:	15-11239
Software Project Manager:	Maarten Versteeg
Start Date Phase C/D:	11/01/2005
End Project:	2/18/2008 (instrument delivery to GSFC)

1.2 Project Overview

Southwest Research Institute develops a UV spectrometer instrument LAMP for flight on the LRO mission. The LRO project is part of the Robotic Lunar Exploration Program and is being developed by GSFC. The LRO spacecraft is planned to be launched in October 2008, for a basic 1 year mission. The lunar mapping mission will be preformed from a 50 km, 113 minute lunar polar orbit.

The LAMP instrument will be almost identical to the Alice instrument that is being prepared for launch on the New-Horizons mission to Pluto. The most notable extension to the instrument is a Lunar Terminator Sensor (LTS). This sensor is used by the instrument to autonomously detect the lunar terminator passing and switch the instrument between an active and safe mode. Using this LTS is possible to acquire science data close to the lunar terminator without very precise orbit knowledge.

1.3 Document Overview

This document contains the software users manual for the LRO-LAMP instrument. Although it is in principle a software users manual, it describes many of the instrument operational systems aspects as the software determines many of these. The document focuses on the software referred to as the flight software (LAMP Flight Software - LAFS) but were needed it will also include some aspects of the LAMP GSE system and its software (referred to as the ground software: LAMP Ground Software - LAGS). In some cases, operational issues are mentioned that may originate from the spacecraft or spacecraft GSE level.

Chapter 2 of this document starts with a general description of the function and operations of the LRO-LAMP instrument; this provides a framework for the instrument operations. This chapter also describes some operational aspects and the relations between the different parts of the LRO-LAMP system.

The next chapter (Chapter 3) contains an overview of the instrument operations. This chapter describes the general operations and is intended to provide sufficient context to enable understanding of the more detailed description focusing on details that is provided in the next chapter (Chapter 4). This should make it possible to understand the descriptions given in that chapter for which no order could be found that would allow for sequential reading.

In a number of appendixes, the detailed description of the telemetry is given including the compete description of the low speed telemetry and the telecommands including parameters and operational effects. Also listed in Appendix F are the contents of the internal stored parameter list, the meaning of the stored entries and their initial and hard-coded default values. The last appendix lists the error codes that are used by the software to report problems in a field in the housekeeping telemetry.

1.4 Related Documents

Document ID:	11239-LAMP-SDP-01
Originator:	Southwest Research Institute, San Antonio TX
Issue:	Release 1, January 2006
Title:	Software Development Plan for LRO-LAMP Flight and Ground Software
Applicability:	Describes the processes and procedures used for software development of the PERSI-LRO-LAMP flight and Ground software.

Document ID:	11239-LAMP_SRS-01
Originator:	Southwest Research Institute, San Antonio TX
Issue:	Revision 0, March 2006
Title:	Software Requirements Specification for the LRO-LAMP Flight and Ground Software
Applicability:	Describes Software Requirements Specification for LRO-LAMP Flight and Ground software.

Document ID:	11239-LAFS_SDD-01
Originator:	Southwest Research Institute, San Antonio TX
Issue:	Revision 0, Change 0
Title:	Software Description Document LAFS software
Applicability:	Describes as build LRO-LAMP Flight software.

Document ID:	11239-LAFS_VDD-01
Originator:	Southwest Research Institute, San Antonio TX
Issue:	Revision 0, Change 0, February 2007
Title:	Version Description Document for LAFS software version 1.00
Applicability:	Describes as build LRO-LAMP Flight software.

Document ID:	PAIP-05-15-11239
Originator:	Southwest Research Institute, San Antonio TX
Issue:	Revision 0, November 2005
Title:	Performance Assurance Implementation Plan for LRO-LAMP Instrument
Applicability:	Overall PAIP for LRO-LAMP

Document ID:	431-ICD-000106
Originator:	Goddard Space Flight Center
Issue:	Revision – March 2006
Title:	LAMP to Spacecraft Data Interface Control Document
Applicability:	Contains data interface description between Spacecraft and LAMP Instrument

Document ID:	431-ICD-000096
Originator:	Goddard Space Flight Center
Issue:	Revision – March 2006
Title:	LAMP to Spacecraft Electrical Interface Control Document
Applicability:	Contains electrical interface description between Spacecraft and LAMP Instrument

Document ID:	11239-IRD-01
Originator:	Southwest Research Institute, San Antonio TX
Issue:	Revision 0, February 2006
Title:	Instrument Specification for the LRO-LAMP
Applicability:	Contains LAMP Instrument description

Document ID: 11239-CDHSPEC-01
Originator: Southwest Research Institute, San Antonio TX
Issue:
Title: Specification for the LRO-LAMP Electronics
Applicability: Contains specification of the hardware of the LRO-LAMP C&DH

1.5 Acronyms and Abbreviations

Acronym	Explanation
A	Ampere
ACQMEM	Acquisition memory
Alice	Previous UV spectrometer instruments, just pretty a name
AnodeV	Anode voltage
APID	Application ID
bps	bits per second
C&DH	Command and Data Handling
CCSDS	Consultative Committee for Space Data Systems
CM	Command Message
CRC	Cyclic Redundancy Check
DDL	Double Delay Line
EEPROM	Electrically Erasable PROM
EGSE	Electrical Ground Support Equipment
EM	Electrical Model
FIFO	First In First Out (buffer)
FM	Flight Model
GSE	Ground Support Equipment
GSEOS	Ground Support Equipment – Operating System
HK	Housekeeping (data packet)
HS	High Speed (Telemetry)
HV	High Voltage
HVPS	High Voltage Power Supply
HW	Hardware
Hz	Hertz – 1/second
I&T	Integration and Test
I/F	Interface
ICD	Interface Control Document
ISR	Interrupt Service Procedure
ITF	Instrument Transfer Frame
kbps	kilo bits per second (= 1000 bps)
kHz	kilo Hertz = 1000 Hertz

Acronym	Explanation
kV	kilo Volt = 1000 Volt
LAT	Limited Angle Torque (motor)
LS	Low Speed (Telemetry)
LVDS	Low Voltage Differential Signal (high speed interface specification)
LVPS	Low Voltage Power Supply
Mbps	Mega bits per second
MCP	Micro Channel Plate (detector)
McpV	MCP voltage
MET	Mission Elapsed Time
ms	milli second
NH	New Horizons
OAP	Off-Axis Parabolic mirror
LAFS	LAMP Flight Software
LAGS	LAMP Ground Software
P-Alice	PERSI-Alice to distinguish it from Rosetta-Alice (R-Alice)
PC	Personal Computer
PERSI	Pluto Exploration Remote Sensing Investigation
PHD	Pulse Height Distribution
PPS	Pulse Per Second
PROM	Programmable Read Only Memory
RAM	Random Access Memory
s, sec	seconds
S/C	Spacecraft
SMA	Shape Metal Alloy (actuator)
SMM	State Machine Mode (backup acquisition mode)
SOC	Solar Occultation Channel
SRS	Software Requirements Specification
SSR	Solid State Recorder (spacecraft data storage)
STB	Software Test Bed
StripI	strip current
SW	Software
SwRI	Southwest Research Institute
TC	TeleCommand
TiNi	Brand name of used shape metal alloy (SMA) actuators
TM	Telemetry
µA	Micro Ampere = 10^{-6} Ampere
UART	Universal Asynchroneous Receiver / Transmitter

Acronym	Explanation
UV	Ultraviolet
WPA	Wax Pellet Actuator
XOR	Exclusive OR (logical operation)

1.6 Notation Conventions

Wherever a specific reference is made to a Telecommand, a field in the Housekeeping packet (see Appendix C), or a field in the Parameter file (see Appendix F), the parameter name will be preceded by respectively ‘Tc_’, ‘Hk_’ or ‘P_’. Error codes are indicated by their name and are preceded by lowercase letters ‘ec’, all error codes are listed in Appendix G.

2. INSTRUMENT SYSTEM OVERVIEW

This section gives an overview of the LRO-LAMP instrument including operations and describes the different software modules within the LRO-LAMP instrument system and their relations. The LRO-LAMP instrument is part of the LRO investigation, but has an independent electrical and mechanical interface with the LRO spacecraft.

To provide a context for the description of the Software and the interfaces, this chapter starts with an overview of the LAMP Instrument. The LRO-LAMP spectrometer consists of an optical system and the controlling/power electronic hardware. To provide an overview of the spectrometer functions, the operational modes and the produced science data are briefly described. Next an overview of the instrument electrical hardware is provided, as this is needed to understand the functions that the software has to perform. Finally, the interfaces of the flight software to the different hardware components are described.

2.1 LRO-LAMP Instrument overview

An opto-mechanical view of the LAMP UV spectrometer instrument is shown in Figure 1. The instrument is a relatively simple design, consisting of a telescope, a Rowland-circle spectrograph, and an electronics section. The telescope section contains a 42 x 65 mm^2 clear aperture off-axis paraboloidal (OAP) mirror. The OAP collects the incoming light that passes through the entrance aperture at the front end of the instrument and directs it towards the entrance slit of a 0.15-m normal incidence Rowland circle imaging spectrograph. The imaging spectrograph, housed in the spectrograph section of the instrument, contains a reflective holographic diffraction grating and a UV-sensitive, solar-blind, 2-D open-structure micro channel plate (MCP) detector located on the Rowland circle. This MCP detector utilizes a double-delay line (DDL) anode readout array and an MCP Z-stack composed of three MCPs stacked together. The command-and-data-handling (C&DH) and detector electronics (DE) section is directly behind the spectrograph section next to the detector. Dual redundant high-voltage power supplies (HVPSs) that bias the detector MCP stack reside in a housing cavity behind the OAP mirror.

The DDL detector system outputs to the command-and-data-handling (C&DH) electronics, pixel location for each detected photon event consisting of a spectral and spatial coordinate. The first (x) dimension provides the spectral location of the detected photon and the second (y) dimension provides one-dimensional spatial information. The events are processed by the C&DH electronics. The C&DH is also the controller of LAMP instrument; it receives commands from the spacecraft, acquires data from the MCP detector system, and returns telemetry to the spacecraft. Science data generation is performed by the detector hardware but the CDH also controls this function. The LAMP instrument has two acquisition modes (see below) in which the spectral/spatial data from the detector is processed by the C&DH subsystem.

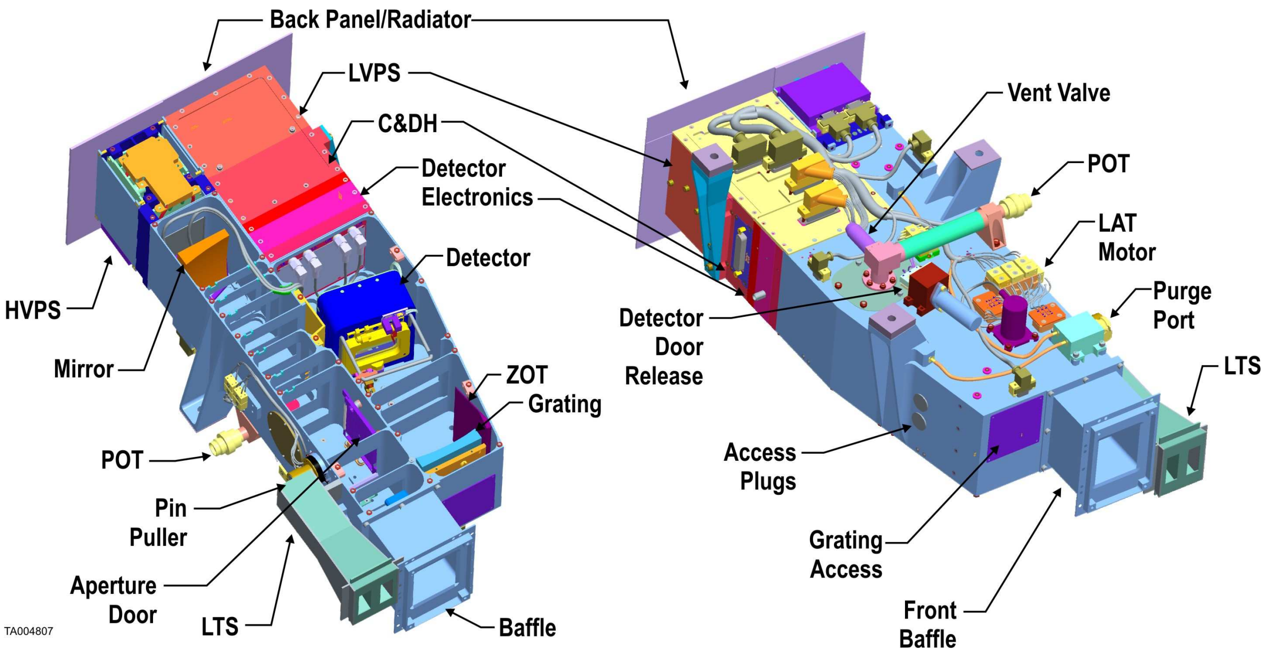

Figure 1: Three-dimensional view of LAMP with the top cover and baffle removed.

All following descriptions assume a nominal operating instrument. The instrument hardware provides a basic, hardware controlled, default pixel list acquisition mode which is activated when the instrument control hardware detects (multiple) successive (software) failures. This mode is called the State Machine Mode (SMM), for a description of this mode the reader is referred to the instrument C&DH hardware description.

Both optical elements (OAP mirror and grating) are each a single monolithic piece of diamond-turned aluminum, coated with electroless Ni and polished using low-scatter polishing techniques. The optical surfaces are coated with sputtered aluminum and a protective overcoat of magnesium fluoride (Al/MgF_2) for optimum reflectivity within the far UV (FUV) spectral passband of LAMP (650-1200 Å). Besides using low scatter optics, additional control of internal stray light is achieved by the use of telescope baffle vanes, and a low-scatter holographic diffraction grating with near-zero line ghost problems. The telescope baffle vanes also act to shield the OAP mirror from bombardment of small particles that can enter the telescope entrance aperture over a large range of incoming angles. A zero-order baffle in the spectrograph section prevents zero order light reflected off the grating from scattering back to the detector.

For contamination control during flight, heaters are mounted to the back surfaces of the OAP mirror and grating to prevent cold trapping of contaminants. The MCPs are themselves heaters; when high voltage is applied to the MCP Z-stack, they heat up because of Joule-heating. This property can be utilized during flight to prevent cold-trapping of contaminants on the MCP surfaces. In addition, the detector has a vacuum cover with a MgF_2 window to protect the MCP and the CsI photo cathodes from moisture exposure during ground operations and integration. This once-acting door will open at a suitable time after launch when the spacecraft has had an adequate period of time for outgassing.

The housing structure of LAMP is machined from a solid piece of 6061-T651 aluminum, chosen because of its strong, lightweight properties. In addition, aluminum allows for an athermal design, since both optical elements (i.e., the OAP primary mirror and grating) are also made of aluminum.

To protect the interior of the instrument during ground operations and test, a front aperture door that can open and close by telecommand is included in the LAMP design. This door is housed in the telescope section of the instrument. To further protect the interior of the instrument during ground handling and testing, a red-tag cover will be used that attaches to the front end of the telescope section.

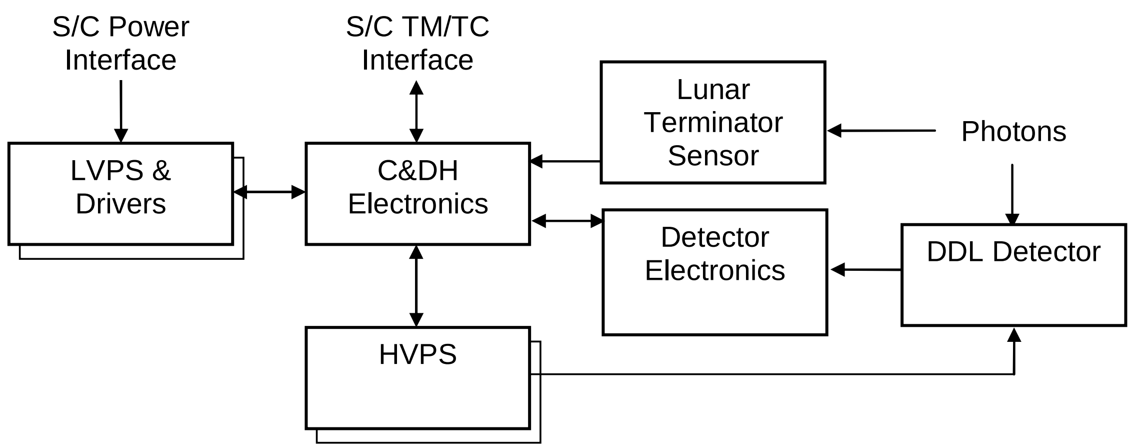

Figure 2: LAMP instrument top level electronics block diagram

All following descriptions assume a nominal operating instrument. The instrument hardware also provides a basic, hardware controlled, default pixel list acquisition mode, which is activated when the instrument control hardware detects (multiple) successive (software) failures. This mode is called the 'hardware limp-along' mode; for a description of this mode, the reader is referred to the instrument C&DH hardware description. Once this mode is activated any software control of the acquisition operation is disabled, until the instrument is reset (power cycle or S/C reset).

2.1.1 Science Operations

2.1.1.1 Detector and Detector Electronics

The LAMP instrument is a Rowland-circle imaging spectrograph that uses state-of-the-art MCP detector and UV optics technology. The telescope section focuses light entering the entrance apertures onto the entrance slit of the spectrograph section. Light that passes through the slit strikes a concave holographic diffraction grating, which focuses the LAMP passband wavelengths onto the MCP detector. Photoelectrons are created by the photocathode material on the front surface of the MCP Z-Stack and are multiplied by the Z-Stack to create an amplified charge cloud of ~10^7 electrons/event that is accelerated across a narrow gap to the DDL 2-D anode array. The multiplication factor (or gain) of the detector depends on the supplied detector high voltage. This allows the HVPS setting to be used as a gain control. When this charge cloud exceeds a specified amplitude level set by the detector threshold, it will result in a valid detected event. In response to the event, the anode outputs signals that are converted by the detector electronics into a pixel location on the array.

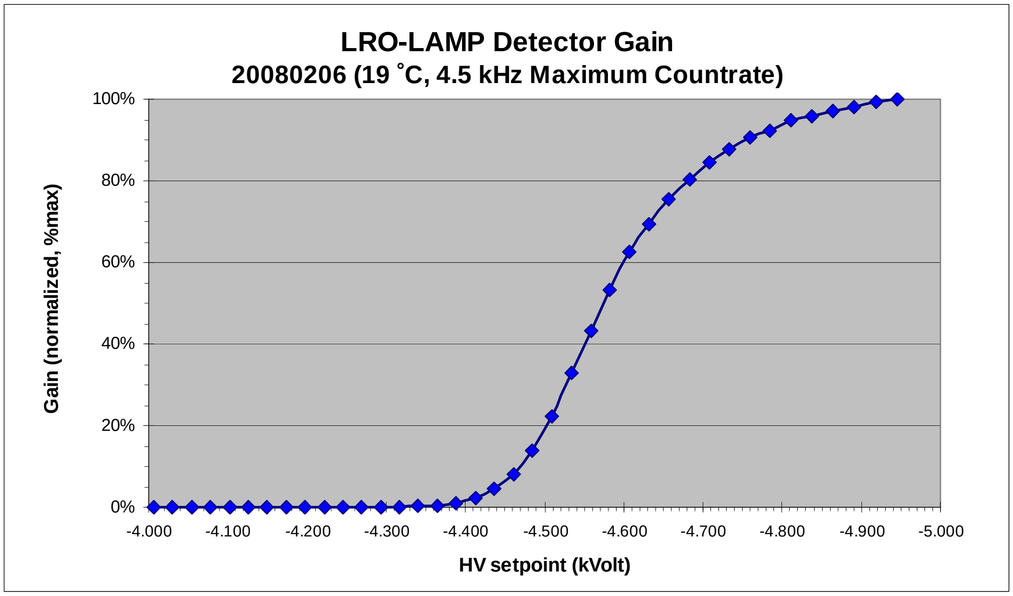

Figure 3: LAMP detector multiplication factor (gain) dependant on High Voltage

Thus the use of the word pixel refers to the resolved location where the charge cloud hits the anode. The detector electronics encodes 1024 spectral pixel columns and 32 spatial pixels. For each event, the detector also reports the amplitude of each charge pulse as a six-bit number that can in histogram mode be histogrammed into a pulse-height distribution by the C&DH electronics. The field-of-view (FOV) of the instrument is such that six of the spatial rows (3 top/3 bottom) are not in the active field of view of the instrument and are used to store the pulse-height data along with two detector electronic stimulation pixels, both of which are used for electronic diagnostics and detector health monitoring.

The LAMP instrument can process the detected pixel events in two different primary modes of operation – pixel list and histogram which are described below. In both acquisition modes, the hardware provides a double buffer, one buffer is used for the current data acquisition, and the data in the other buffer is sent to the spacecraft. This makes it possible to perform continuous acquisitions. In both cases science data transfer from the instrument to the spacecraft will occur at a burst rate of 1 Mbps. The block size of the transfers is 32 kword, resulting in a transfer duration of about ½ second.

2.1.1.2 Pixel List Acquisition Mode

The main LRO-LAMP data acquisition mode is the Pixel-list acquisition; this mode is used for lunar observations. In pixel list acquisition mode, the instrument stores event data as it arrives from the detector system. Each detected event is stored in a consecutive word (16 bits) in the acquisition memory. In this mode the detector memory is used as a 32,767 locations deep list that can store 32,767 events. To indicate event information, the top bit of the word is cleared and the remaining 15 bits contain the event information. In order to provide timing information, “time hacks” are also stored in this acquisition memory. These time-hacks are inserted at a programmable interval between 4 and 512 ms, and also take up one word in the detector memory. For the time hacks the top bit is set and the lower 15 bits provide a counter value. During a pixel list acquisition, the pulse height data is discarded. When the acquisition memory is full or the acquisition is stopped, the buffer contents is read out and a block of 32 k words are sent to the spacecraft. The filling rate of the acquisition memory depends on the actual event rate and the selected time hack interval.

2.1.1.3 Histogram Acquisition Mode

In addition LRO-LAMP provides a histogram acquisition mode. This mode is mainly used for calibration acquisition operations and allows for long integrations to obtain observations of dim objects with high signal to noise ratios. In histogram acquisition mode, the detector counts the number of events that are occurring at each of the 1024*32 = 32,768 (32 k) "locations" provided by the detector electronics over a period of time (called an observation). Each of the histogram counters consists of a 16-bit number hence, for each location a maximum number of 65,535 events can be counted. When this maximum number is reached during an observation, the counter is no longer incremented and the corresponding pixel becomes saturated. By operationally selecting suitable exposure durations (excessive) saturation may be prevented. In addition to the spectral/spatial histogram, a pulse height histogram based on the pulse height information that is provided with each detected event is acquired. For histogram operation, a small area in the bottom row is reserved and 64 locations are used to store the counter for this histogram. This pulse height histogram is referred to as the pulse height distribution, and is useful for instantaneous health monitoring of the detector MCP gain characteristics. These counters are also 16 bits deep and, thus provides for a maximum count of 65,535 events per bin. When an observation is completed a single read out results, in which each of the 32 k histogram locations is represented as a 16-bit number.

2.1.1.4 Science data format and masking operation

The LAFS does not have direct access to the detector event data; this data is handled by the acquisition hardware and stored in the acquisition memory according to the selected acquisition mode. Each detector event consists of an event location specified as a spectral (10-bits resolution) and a spatial (5-bits resolution) coordinate in addition to this the 'pulse height' of an event is determined (6-bits resolution). The acquisition mode determines how this data is stored in the acquisition memory:

- In pixel list mode, the acquisition memory contains addresses of pixel events in sequence as they occur. In Pixel List mode there is also a time mark (a special value) inserted into the sequence on a periodic basis in order to provide event timing information.
- In histogram mode, there is a mapping between locations in the memory array and pixel locations on the detector. Each location is associated with one pixel, and contains the total number of event counts for that pixel that has accumulated within the integration period. A small area of the memory (64 entries) is set aside to count the number of pulse heights for each intensity, resulting in a pulse height distribution.

Before any event is processed, a 'hot-pixel' filtering operation is performed. This filtering operation allows for the exclusion of events from certain areas of the detector array. Eight user definable filter masks are provided, and one fixed filter mask is defined that masks the lower left corner of the detector (this area contains the header word and in histogram mode the pulse height distribution data.) The hardware provided masks consists of rectangular areas with a size of 4 spatial (y) rows and 32 spectral (x) columns. Events that are registered within this area are excluded from any further processing, but these events are included in the measured (raw) event countrate. The location of each of the 8 user definable masks is specified using a single byte. The top 3 bits of each mask specification define the spatial location of the mask and the bottom 5 bits specify the spectral location. This means that the rectangular filters can only be located at multiples of their respective spatial (y) and spectral (x) sizes. The filter area starts in the specified row and column and run up to (but not including) the next possible filter start row and column.

Dimension	Data range	Hot Segment Filter/Mask			
		specification byte	start location	possible start location	end location
Spatial (y)	**0-31**	**top 3 bits**	**4*y**	**0,4,8, … 28**	**4*y+3**
Spectral (x)	**0-1023**	**lower 5 bits**	**32*x**	**0,32,64, … 992**	**32*x+31**

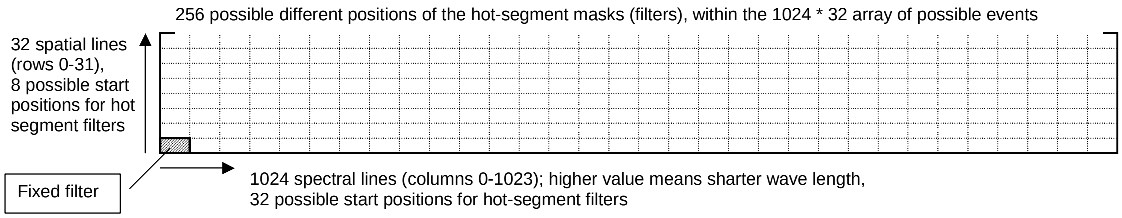

Figure 4: Hot pixel mask (filter) locations.

Upon system (hardware) reset, all user definable filter specifications are reset to zero, so the location of the user definable filter areas correspond to the fixed filter in the lower left corner of the event array.

2.2 Software Test and Communication Configurations

The complete developed software suite for the LRO-LAMP project consists of two main parts. One of these is included in the LRO-LAMP instrument, and the other one is located in the LRO-LAMP GSE engineering workstation.

During the LRO-LAMP development and flight, different communication configurations have to be supported. Final configuration will be the flight configuration in which the instrument is mounted on the spacecraft and the spacecraft provides a TC/TM communication with the ground system. During the mission when the spacecraft is in lunar orbit the spacecraft will not be in constant contact with the ground system, but periodic contacts are foreseen. During these contact periods, command sequences will be uploaded and telemetry including science data will be downloaded. Outside these contact periods the spacecraft will operate autonomously based on command sequences.

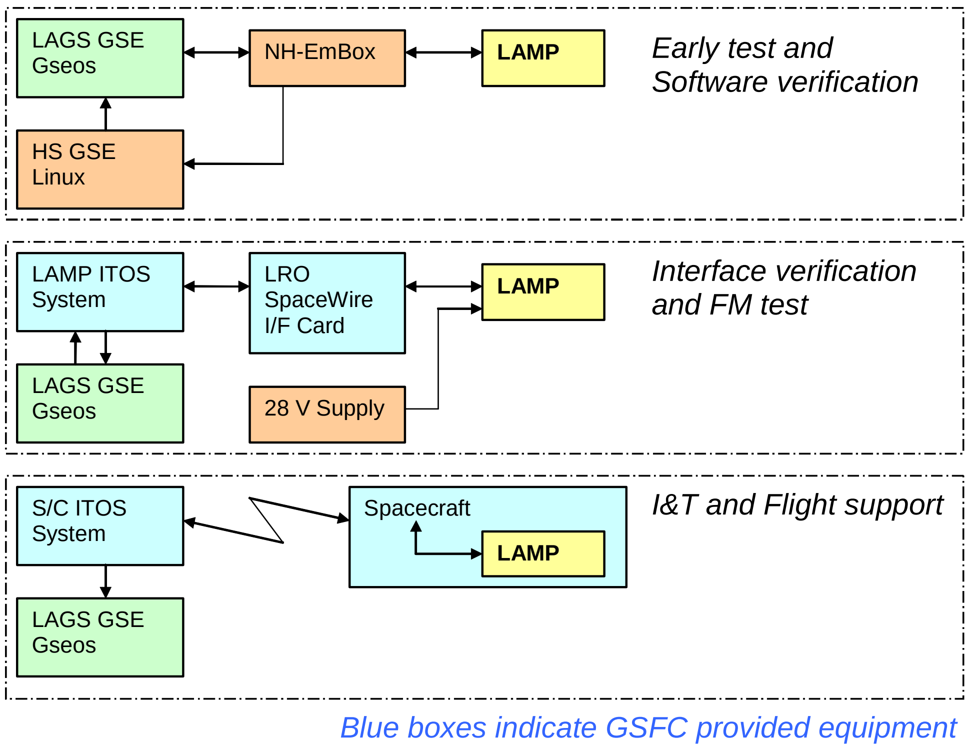

Figure 5: Test and communication configurations for the LRO-LAMP system

During development, test and integration at the various levels, the communication between the LAMP ground system and the LAMP instrument will use various hardware configurations as shown in Figure 5. These different configurations support the different test and integration activities.

In addition to the available hardware platforms, a software simulator of the processor (Keil51 simulator) is available to support the mission. This simulator includes capabilities to functionally simulate the hardware that directly interfaces with the processor; this provides a basic environment for the execution of the flight software. This simulator environment will be maintained as it provides a platform to support tests and it provides a very detailed visibility of the internal states of the processor. This simulator together with a software breadboard system (LAMP Instrument Interface Simulator (LIIS)) will form the basis of the mission support system.

2.3 Hardware Operational Environment

2.3.1 Processing Hardware

Following is a summary of the hardware environment in which the LAMP Flight Software executes. Refer to Appendix A for an overview and the LAMP C&DH specification for details about LAMP hardware.

Here are the primary features of the LAFS execution environment:

- 8051 compatible micro controller, 4.000 MHz clock,

- 32 kilobytes non-volatile memory: primary program memory (PROM),
- 4*32 kilobytes non-volatile memory: patchable program and tables memory (EEPROM),
- 32 kilobytes volatile memory: general data memory (RAM),
- Circuitry for receiving telecommands (redundant TC interface), with external interrupt,
- Circuitry for sending telemetry (redundant low rate TM interface), with external interrupt,
- Circuitry for real-time timekeeping; including periodic and timesync interrupt,
- Science data acquisition memory (2* 64 kbyte(used): ACQMEM) and control circuitry, also generating the high rate science TM packets,
- Detector interface (including event counter, STIM etc.),
- Watchdog timer,
- Circuitry for acquiring analog housekeeping (status) measurements,
- Circuitry for controlling the system actuators (including heaters).

2.3.2 Instrument Internal interfaces

An overview of the interfaces of the P-LAMP flight software is presented in the software context diagram shown in Figure 6.

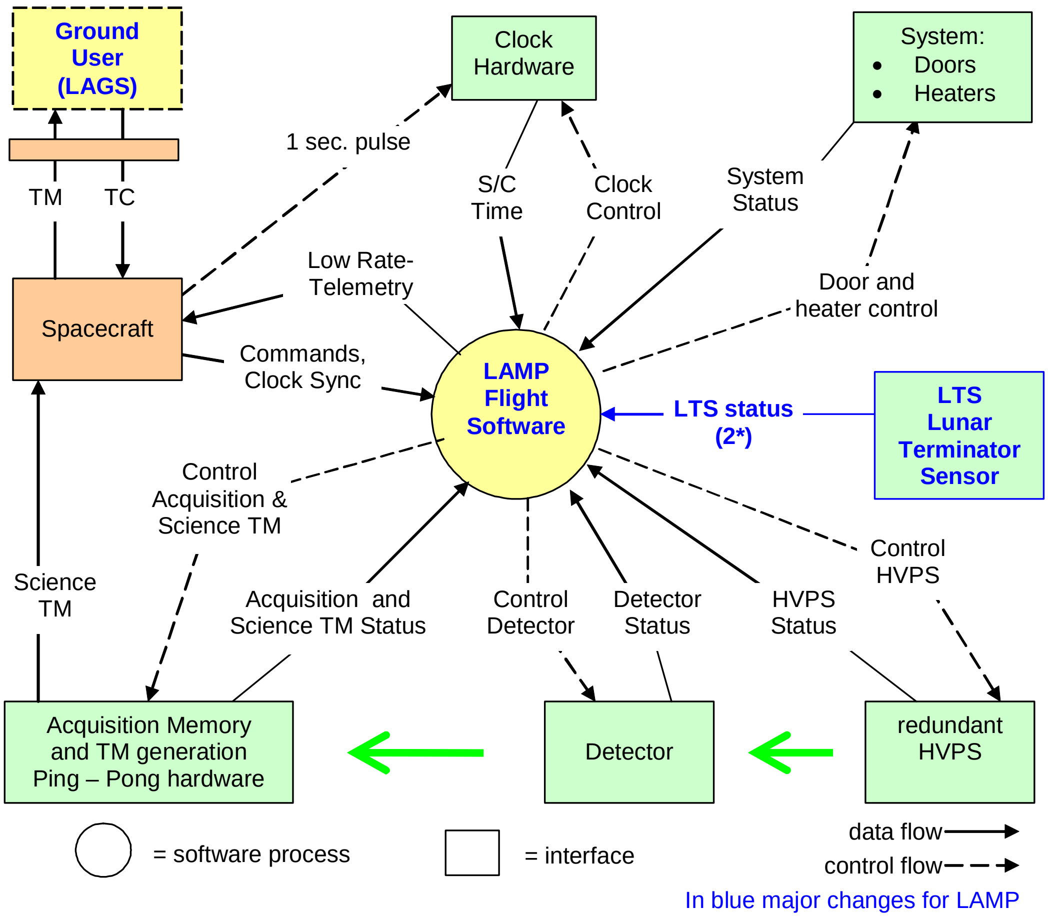

Figure 6: LRO-LAMP Flight Software Context diagram

2.3.3 Safety Configurations

For use during ground testing there are two redundant hardware safing mechanisms included in the LAMP instrument. These safing mechanisms define an operational mode, which is configured by means of an external instrument red-tag safety plug. This Actuator Safety plug will allow for separate enabling of the primary and redundant sides:

- High Voltage safing plug (separate for primary and redundant HVPSs)
- Actuator safing plug (separate for primary and redundant sides)

The four safing status signals are read by the LAFS to be reported in the HK data stream to allow verification during ground testing. This status is only reported; it does not influence any (on-board) software control function.

2.3.3.1 HV safety configuration

The LAMP instrument has a hardware selected safe operational mode in which the HV activation is limited to a HV value that allows safe operation of the MCP detector at any pressure. This prevents any damage to the instrument during integration and test caused by unintentional activation of the HVPS (HV activation to voltage levels exceeding 1 kVolt at room pressure can cause arcing of the detector, but at some partial pressures arching can occur at even lower voltages). This operational mode is configured by means of an external instrument red-tag safety plug, which provides a separate control of either of the redundant HVPSs.

This HV safing mode reduces the full-scale command signal for the HVPS. This HV safing mode does not affect the HV read back values, so these will read back at the corresponding limited range values (i.e., read back values correspond to the actual voltage, not the reduced value). This means that the HV safety checking cannot be activated in the HV safing mode, as the read backs will show the limited values:

- Mcp Voltage read back of the scaled down value, so the check verifying commanded versus read-back HV will fail,
- Strip Current linear with Mcp Voltage results in read back of scaled down value,
- Anode Voltage has a nominal operational value of about 600 Volt, at the reduced control value of the HVPS the Anode Voltage regulation will not yet be active, so a reduced value will be read**.**

A consequence of these differences is that when the HV safety plug is in place nominal systems operations are not possible. The 'Mcp Voltage' and the 'Anode Voltage' safety mechanisms will detect non-nominal conditions and trigger the safety mechanism. To operate LAMP (for instance to verify commanding sequences), the “Mcp Voltage” and the 'Anode Voltage' safety checks must be disabled.

2.3.3.2 Door actuator safety configuration

The LAMP instrument has a hardware selected (configuration plug) safe configuration in which one time door and valve actuators physically cannot be activated. A separate safing for primary and redundant actuators is available. This prevents these one-time mechanisms from being activated unintentionally during integration and test. Note that this safety configuration does not affect the aperture door or heaters from being operated (the aperture door will need to be unlocked).

2.4 Instrument External (spacecraft) interfaces

Although the LAMP instrument has six separate independent redundant interfaces directly to the spacecraft.

- Instrument power interface
- Time synchronization interface
- Low speed serial telecommand interface (TC)
- Low speed serial telemetry interface (TM)
- High speed science telemetry (HS)
- Instrument reset interface

2.5 GSE system overview

The initial LRO-LAMP Ground system configuration will be re-using the New Horizons-LAMP GSE system. This initial LRO-LAMP Ground system will consist of two PCs (see **Error! Reference source**

not found.) with a custom hardware unit that constitutes the spacecraft interface emulator. This emulator will mimic all the redundant, physical high speed, low speed and power interfaces from the spacecraft to the instrument. One PC, the Science Acquisition PC (PSAP), will handle the high rate science TM stream from the instrument. This PC will be running under the Linux operating system and will only perform this high-speed acquisition function. The received science data is stored in files that can be accessed /distributed through a network interface.

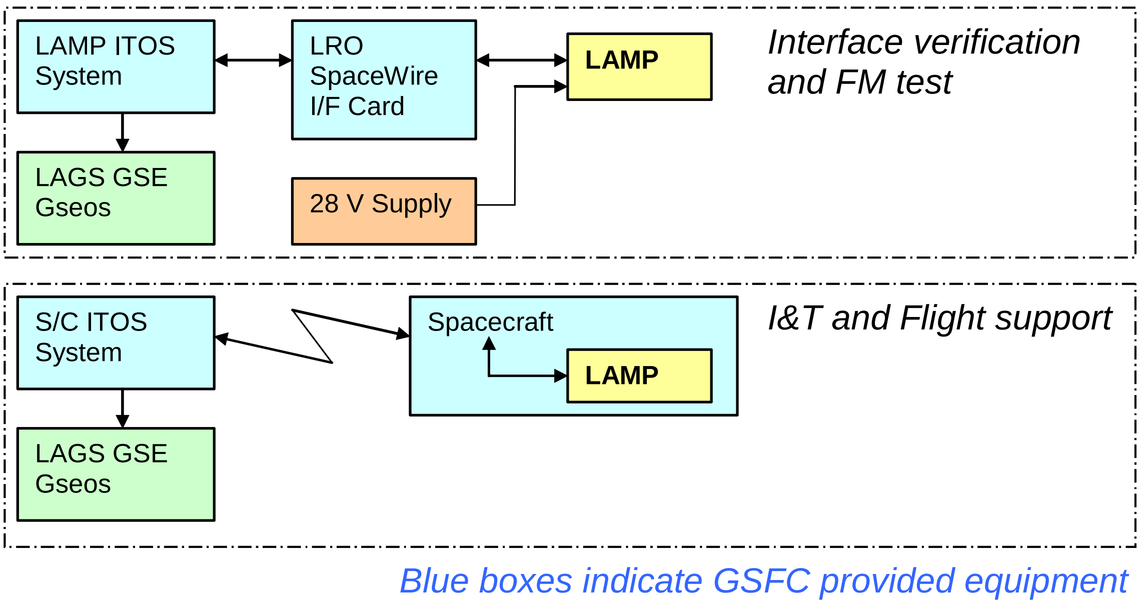

Figure 7: Overview LAMP GSE system (Engineering support, Science processing separate)

In both the ITOS based test configurations are shown. Although both configurations probably never will exist in parallel connected to the same network this figure is used to indicate that both configurations are identical from an instrument GSE point of view. Actually the right configuration is used to represent all configurations after the LAMP instrument is integated on the S/C as they are identical from the LAMP instrument GSE point of view. The LAMP GSE system is identical in both cases and receives the LAMP related telemetry via a TCP/IP based network connection. This LAMP GSE is based on the GSEOS framework that was developed for the Rosetta and new-Horizons LAMP instruments.

3. INSTRUMENT OPERATIONS OVERVIEW

3.1 Instrument Startup

Upon power on or software reset, the software starts and performs a code check. Instrument power may be applied either to instrument power bus A, bus B, or both. The code check determines which code will be started, either PROM or one of the four EEPROM pages. If no problems are detected in the code check of the PROM, execution initially starts in PROM. After completion of this code check and clearing of the memory, the various software modules are initialized; this basic software initialization includes:

- Enter safe state (deactivate any actuator operation)
- Pre-set parameter table with hard coded (nominal operational) values
- Parameter table initialization (with majority vote based values from EEPROM)
- Set discriminator to default value (based on parameter table)
- Start the aperture closing operation (if enabled and actuator power available)

After this, the nominal operations of the system start and the software starts generation of the telemetry data stream at a rate of one packet per second. The first telemetry packet will appear between 2 and 3 seconds after startup depending on whether an external sync pulse (1PPS) is detected during the system startup.

The LAMP instrument can be in one of four states (including powered off) at any time. Three of these states are actually 'operational' states. Initially, after power up or a reset, the instrument will always enter the Safe state. In the safe state the instrument will always be idle, the HVPS, heaters and actuators will all be deactivated.

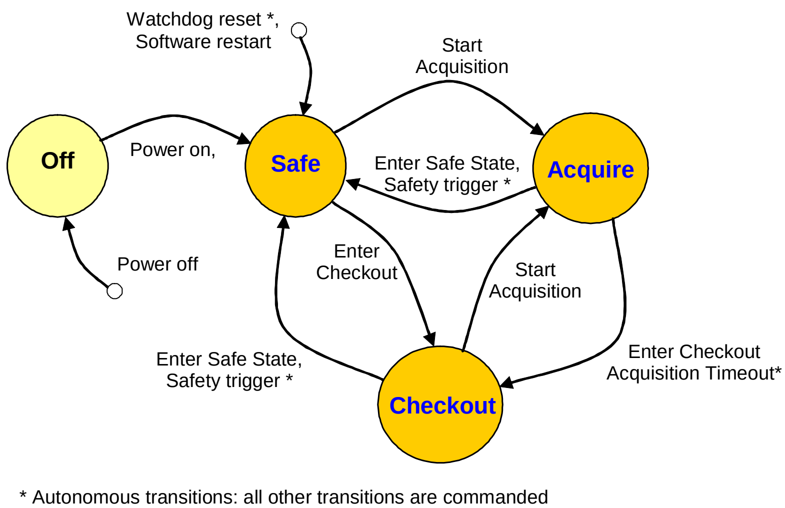

Figure 8: Operational states, state diagram

Changes between the states can either be the result of received telecommands or autonomous by the instrument due to an unmasked safety trigger. The current state of the instrument is always reported in the housekeeping telemetry.

3.2 Telemetry Generation

Every second after startup in nominal situation, the instrument will generate a single low speed telemetry packet. The gathering of the data for the telemetry packet starts immediately after the instrument receives an external sync pulse (1PPS). If no 1PPS pulse is received by the instrument, the instrument will continue to generate a single telemetry packet every second, now based on the internal instrument clock. The gathering of the information either retrieves information from the hardware directly (where possible) or in some cases retrieves most recent acquired data (for measurements taking more time or periodic measurements). This means that the reported housekeeping data may be up to 1 second old.

In each telemetry packet, the instrument reports the instrument concept of Spacecraft Time (MET). This time is represented as a 32-bit integer number specifying the number of seconds that have passed since some epoch. A 32-bit integer number allows the specification of a period of $2^{32} = 4294967296$ seconds (about 136 years). Normally this instrument time will be synchronized to the spacecraft time and allow for correlation of the instrument reported status with spacecraft events.

The generated telemetry packet is transferred to the spacecraft in an ITF (Instrument Transfer Frame) format. The generated ITF always contains a complete CCSDS packet containing the instrument housekeeping (APID 129 (=0x81)). In addition to this, the LAMP generated ITF may contain a memory dump packet. This memory dump packet is again a CCSDS formatted TM packet (APID 130 (=0x82)). The instrument housekeeping packet contains the complete instrument state description including the operation mode. A single packet was defined that includes both the operational and the diagnostic information needed during normal operations of the instrument. Some additional diagnostic data may be obtained by directly dumping selected instrument memory areas using the available memory dump command.

In addition to this low speed telemetry, the instrument uses a high-speed interface to transfer science data from the instrument to the spacecraft. This transfer consists of raw science frames which are always 32k word (16-bit words = 65536 bytes) in the LAMP case. These packets will be stored in the spacecraft Solid State Recorder (SSR) memory file each frame will be preceded by a 6 byte Spacecraft Time value indicating when the frame was received from the instrument. These files will be transferred to ground using a CCSDS based file transfer protocol. High-speed science data frames are only produced during or at the end of commanded acquisition operations. Each generated science frame includes a sequence number counting the number of frames generated since the last instrument startup; this number is also included in the housekeeping telemetry stream allowing for correlation between the generated science data and the housekeeping data describing the instrument conditions during the acquisition.

3.3 Telecommanding

The instrument operations are commanded using a set of 25 separate telecommands. The instrument can process up to one command per second. The acceptance and completion status of the command execution is reported in the housekeeping data. The instrument verifies incoming telecommands before they can be executed; this basic verification includes a format and a checksum check of the telecommand. The instrument has two redundant telecommand interfaces, but, within any given one second timeframe, the instrument will only accept telecommands from a single interface. Based on data received from both interfaces and the status of the also redundant 1PPS synchronization signal, the instrument decides which telemetry interface to use.

In addition to this verification mechanism, the instrument implements two additional mechanisms to protect the instrument from anomalous telecommands. Some commands are only allowed when the instrument is ‘not in the SAFE state’. In addition to this, a number of commands have been declared ‘critical’. For LAMP this means that within a nominal 30-second timeout period a specific confirmation command has to be received before the actual (critical) command execution starts. During most of the in flight operations, this timeout is short and means that the confirmation already has to be issued before confirmation of the acceptance of the command has been received on the ground.

The set of telecommands can be divided into three categories:

General operations – These allow for the complete basic operational commanding of the instrument. This includes setting and storing of parameters and starting and stopping of the science acquisitions. This set of seven commands allows for the full science operations of the instrument.

Manual operations – Additional capabilities needed during commissioning and instrument verification are provided by 15 additional telecommands that allow for extended command options. Some of these commands may be used for science operations depending on the selected operational philosophy.

Memory functions – Software code management and maintenance and additional debugging functions are provided by three general purpose memory functions that allow for verification, load and dump of memory blocks.

Table 1: Command execution checks

Action	Checks	Effect
Command acceptance	• Command format and checksum verification, • Allowed command state, • Critical command confirmation, • Command parameter verification	Increment either command accepted or command rejected counter, Report error code if command was rejected
Command execution	• Check commanded results, • Completion of command execution	Increment command executed counter when execution is completed successfully, Report error code if command execution failed

Whenever the instrument detects errors while accepting or executing commands, an error will be reported in the generated telemetry packet. This includes an identifier for the telecommand (if available) and a general error code. The error code remains being reported in the telemetry data until another error is detected or the instrument is reset. This simple form of error reporting is limited to reporting a single error per second. An additional mechanism implementing a small error log is available for more extensive problem investigation. The command code for any successful command is also reported in the telemetry data so the telemetry registration can be used to reconstruct the received telecommands. Note that the parameters of a telecommand are not included in this reporting.

3.3.1 Acquisition commanding

The basic science data acquisition function of the instrument is handled by two 'start Acquisition' telecommands. One starts the histogram acquisition mode and the other the pixellist acquisition mode. These two commands each have two parameters; one defines whether the aperture door of the instrument is to be opened for the acquisition, and the other parameter allows for the selection of some test modes. All other acquisition parameters are controlled by the parameter list. This is a list retrieved from non-volatile instrument memory at startup. Once these general instrument parameters have been set, these parameters are expected to need little change, and this mechanism results in a very simple acquisition commanding.

Once the acquisition start has been commanded, the instrument goes through an acquisition startup sequence as listed in Table 2. A number of the steps in this sequence are optional as they depend on the current state of the instrument. This means that the duration of the acquisition startup (although deterministic) depends on the current state of the instrument. In any case, the actual acquisition startup will start synchronized to the spacecraft synchronization signal (next 1PPS synchronization pulse), so the exact time of acquisition start is known and can be planned for. For some cases, it still may be useful to configure the instrument before issuing the start acquisition command.

Table 2: Acquisition startup configuration sequence

Step	Parameter	Duration
Set the discriminator level	P_DISCRIMINATOR	~ 0 seconds
Command the pixel STIM	P_Stim_Enable	~ 0 seconds
Set the pixel list hack interval counter (only used for pixel list acquisition)	P_PIXELLIST_HACK,	~ 0 seconds
Set all eight hot segment masks	P_HOTSEG_1-8	~ 0 seconds
Command the aperture door to the state specified in the command parameter, wait for the control period if the indicated state of the door is not equal to the requested state	telecommand parameter	Door control period or ~ 0 seconds (if door indicated already in requested state)
Command the HVPS to the specified level () following the ramp-up algorithm specified in the HVPS module, the setup will only continue when the specified HV level is actually commanded (reached). If the HV was already commanded on before the start acquisition command this step will not add any time to the acquisition start processing	P_HV_LEVEL	Ramp-up time or ~ 0 seconds (if HV supply already at configured level)
Clear acquisition memory	N/A	~ 0.1 seconds
Initialize the acquisition timeout timer	P_ACQ_TIMEOUT	~ 0 seconds
For histogram acquisition initialize the exposure timeout timer	P_HISTO_EXP_DUR	~ 0 seconds

Whether the aperture door is opened for a specific acquisition depends on the operational procedures. The aperture door could (remain) closed for SOC acquisitions to prevent any spurious light from entering the optical system through the airglow aperture. For airglow acquisitions, the aperture door has to be open; this could either mean that the door is left open all the time or only commanded open for the airglow acquisitions. In either case in order to move the aperture door one of the actuator busses should be active; otherwise, any door commanding will have no effect.

For pixellist acquisitions a science data frame will be generated whenever the acquisition memory if full of data (photon and timehack events). The time it takes for this memory to fill depends on the selected timehack rate and on the brightness of the observed object. When an acquisition memory buffer is full, the acquisition will immediately continue in the next memory buffer while the filled buffer is transferred to the spacecraft and stored in the SSR. As this buffer full event is related to the number of detected event, it is not synchronized to the instrument time sync. The transfer of the first buffer to the spacecraft will be completed in about 0.5 seconds so in nominal operations the buffer will be available for acquisition before the second buffer is filled. For histogram acquisitions there is no such acquisition buffer full status, and the acquisition will continue until a pre-defined exposure period has expired, then the same action takes place as for the pixellist acquisition and the acquisition continues in the second buffer while the first frame is transferred.

Nominally, the acquisition operation will continue until the instrument is commanded to ‘stop acquisition’ explicitly. In addition, a timeout mechanism is implemented that will also terminate the acquisition mode when a predefined timeout period has expired; this mechanism may also be used to command acquisitions of a pre-determined duration. When the acquisition is terminated, the data already in the acquisition buffer is send to the spacecraft. For a pixellist acquisition still, a complete 32 k-word science frame is used even though it may only be partially filled. The used part in the pixellist frame can be recognized either in the pixellist frame itself as the data consists of all ‘zero’ events which will normally not occur in the data and will also be reported in the housekeeping packet. To command the ‘stop acquisition’ either the ‘enter safe state’ or ‘enter checkout state’ telecommand can be used. Both commands have the same effect in that the acquisition is terminated. The ‘enter safe state’ will, in addition to terminating the acquisition, also deactivate the HVPS and close the aperture door (if enabled and actuator power available).

3.3.2 State Machine Mode (SMM)

The State Machine Mode (SMM) is a backup pixellist acquisition mode that is activated autonomously if the software repeatedly fails. It is created to recover from a catastrophic software or system error and is inherently more reliable as it doesn’t need any software to function and only a limited part of the hardware is used. The SMM mode uses hardware defined acquisition parameters that were fixed at instrument integration time. Therefore these parameters probably won’t be the optimal parameters for the acquisition but it gives a chance to recover at least part of the science.

The instrument software is protected from end-less loop by a hardware watchdog mechanism. When this watchdog expires multiple times without the instrument being power cycled or hardware reset, the instrument will enter the hardware controlled State Machine Mode (SMM) acquisition. In this mode the instrument will perform a continuous pixellist acquisition with hardware defined parameters and generate the corresponding pixellist science data frames on the high speed link. The SMM will only terminate when the instrument is switched off. Note that any expiration of the watchdog counter already indicates a non-nominal event; normal instrument operations should never result in expiration of the watchdog timer.

Table 3: State Machine Mode triggers.

Watchdog resets	Minimum duration	Condition	Effect
>=4	16 sec	Always	Request instrument power cycle via HK flag
>=8	32 sec	Double instrument power supply	Activate hardware controlled SMM acquisition
>=16384	18.2 hr	Single instrument power supply	Activate hardware controlled SMM acquisition

After entering the hardware controlled SMM pixellist acquisition mode, the software is locked out from any acquisition control. Software could still be executing though but is will be limited to monitoring hardware operations. In addition telemetry could still be generated which may instrument status (via the low speed telemetry link), but this is no longer required for the acquisition to continue. The hardware controlled acquisition performs a pixellist acquisition with fixed parameters. These parameters were defined at instrument integration time and are listed in Table 4. Activation of the HVPSs will not follow a controlled rampup, the HVPSs will immediately be commanded to the full operating level.

Table 4: State Machine Mode (SMM) fixed operational parameters

Parameter	Fixed value	Remarks
Commanded High Voltage Level	175 (~ -4.95 kVolt)	HW configured value
HVPS select	both	Actual HVPS activation depends on the status of the instrument power busses
STIM	on	
Discriminator Level	43 (~ 0.5 Volt)	HW configured value
Time Hack rate	4 ms	

This watchdog mechanism protects the instrument from a fatal software failure by still providing a limited acquisition mode. Operation software may trigger this mechanism intentionally by causing the watchdog timer to expire multiple times. This provides for a way to test the SMM mode acquisition. This mechanism will fail to protect from a failing command channel; whenever the commanding channel is lost the software may still operate perfectly but just not receive any commands. But this requires a failure of both redundant Telecommanding channels.

4. INSTRUMENT OPERATIONS DETAILS

4.1 Instrument Startup

After power up or reset, the software initially starts executing special code from PROM that based on checksum verification determines which code is going to be executed. The system has redundant code storage in the form of one PROM memory page and four EEPROM memory pages that can all contain a complete executable code image. No copying of code is involved as the code can be directly executed from any of these five pages using page-switching hardware. No further selftest is performed at startup, as no options are available to take corrective actions. A selftest function is available in the normal operational system that allows an operator to select various test functions. Only after completing this code, the operational code is activated.

4.1.1 Startup Code Check

The system has redundant code storage in the form of one PROM memory page and four EEPROM memory pages that can all contain a complete executable code image. The startup implements the startup sequence as listed in Figure 9; this code attempts to find the first code page that contains correct code (starting with PROM and proceeding through the four EEPROM pages if needed). If all five-code storage pages are flagged incorrect, execution starts in the PROM page as this has the least probability of being corrupted.

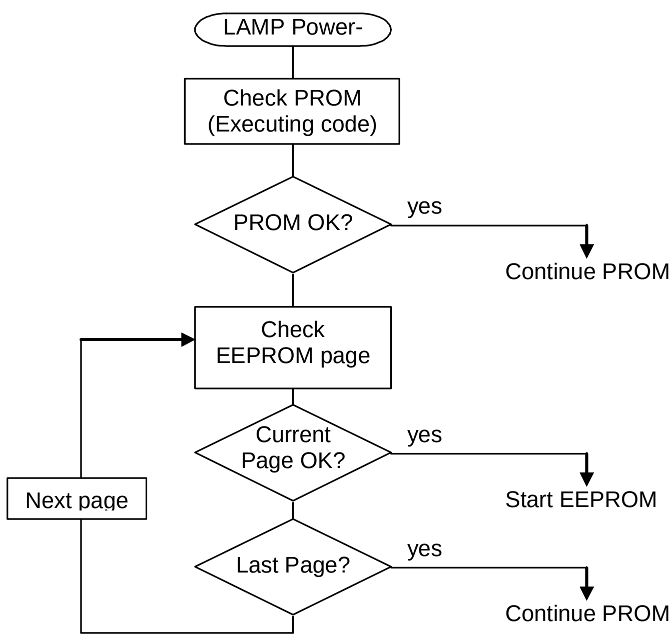

Figure 9: LAMP Startup Code Check

The initial startup software first calculates the checksum over the complete PROM code that is going to be executed. This check uses a byte-wise16-bit xor-rotate checksum (same algorithm as show in section

4.7) resulting in a 16-bit checksum. Each of the four EEPROM pages stores it's own checksum (which could be different when different code versions are loaded). The checksum is calculated over the complete 32-kbyte code memory excluding the last 130 bytes. The first two bytes of these last 130 bytes are used to store the checksum itself (the remaining locations are used to store the 'non-volatile' parameter lists in the EEPROM memory (pages 1-3) copies and as scratch pad during EEPROM memory testing (page 4)). After selecting the code image to be executed, the startup sequence continues with the initialization of the different memory spaces and the start of the actual (operational) instrument software

This code checking is not foolproof. This verification relies on some code executing correctly form the PROM memory (which must be correct) but as this is a smaller amount of code the changes for corruption are less. To enable the check to run, some PROM code should be able to execute, but the required part is very small (~50 bytes) compared to the complete application software. Also the check might fail because of a bit error in a non-essential code location or in the checksum itself, but, after the system is started in EEPROM, it is always possible to jump back to PROM code. Even though this startup code check uses the less reliable memory as a backup strategy for the PROM, it improves the overall chances that the LAMP software will start. The current executing code (PROM/EEPROM and EEPROM page) is reported in two fields in the housekeeping TM packet.

When modifying EEPROM code (patch command), it is safest to start with the last page and keep the lower numbered pages as backup of the original PROM contents (note: LAMP will be launched with five identical copies of the flight software stored in PROM and EEPROM). These are created using the ‘duplicatePROM’ function, which copies the PROM contents to all four EEPROM pages). The programming of EEPROM can only be performed when the PAFS is executing from PROM (see 4.7).

4.1.2 System startup activity timing

The next two tables show the results from simulations of the system during the startup phase. The results show the time (in seconds) at which various activities occur during the startup sequence in a case where no errors are detected in the PROM memory. If errors would be detected, the startup sequence would take longer as additional operations are required to verify the EEPROM code pages.

After simulation start, the instrument software first goes through the nominal startup sequence consisting of a PROM check and initialization of the memory areas. When this is completed, one of the first actions are a number of early hardware initializations including the termination of any ongoing actuator activation (including HVPS). This is visible in both cases: “1.680 – Aperture control stopped”. This means that the complete internal startup is completed within 1.68 seconds. Next the parameter file is read from 3 pages of the EEPROM memory; this is shown when the software switches the EEPROM ‘memctl’ register. After that, the aperture door is commanded to close as defined in the used configuration file; door control is activated for the (configuration file determined) period of about 1.8 seconds. The two presented cases show the effect of the absence of the 1PPS clock on the system startup. If there is no 1PPS clock, the instrument will, after waiting for some time and when no 1PPS is received, start processing on the internal clock. Note that for the simulated 1PPS active case, the simulation case shows a specific case where the 1PPS happens at the same time when the instrument power is applied. If power up occurs in different phase timing may differ slightly.

The hexadecimal dumps in the simulations show the raw contents of the generated housekeeping packets. These consist of the complete ITF packet with the first synchronization byte changed to 0x 64 for simulation control purposes.

1PPS and clock message active

```
  0.000 - simulation start (start 1pps & time message)
  1.000 - 1pps + time msg 10000
  1.680 - Aperture control stopped (- s)
  1.688 - memctl, switch EEPROM page (p0.0) to (p1.3)
  1.697 - memctl, switch EEPROM page (p1.3) to (p2.3)
  1.706 - memctl, switch EEPROM page (p2.3) to (p3.3)
  1.899 - Start Aperture control close
  2.000 - 1pps, previous not acknowledged
  2.087 - TM transfer started
  2.120 - itf transmitted (116 bytes)
0x00: 64 fa 30 04 6b 00 6d 40 00 00 20 21 00 00 00 fe
0x10: 00 00 00 00 0c 82 c0 00 00 59 00 0f 42 40 20 c3
0x20: 21 00 00 00 00 00 00 ff ff fe 00 00 00 10 00 00
0x30: 00 01 86 ca 01 f7 00 00 00 00 7f ff ff ff 00 00
0x40: 00 00 02 00 00 02 00 00 00 0c 00 00 66 66 68 68
0x50: 78 7a 82 66 00 00 00 10 47 11 e0 48 8a 00 00 00
0x60: 02 00 3f 43 45 f1 f3 f5 f7 f9 fb fd aa ff 20 f8
0x70: 01 33 56 17
  3.000 - 1pps + time msg 10002
  3.087 - TM transfer started
  3.120 - itf transmitted (116 bytes)
0x00: 64 fa 30 04 05 00 6d c0 00 00 20 21 00 00 00 fe
0x10: 00 00 00 00 0c 82 c0 01 00 59 00 00 27 12 20 c3
0x20: 6c 00 00 00 00 00 00 ff ff fe 00 00 00 10 00 00
0x30: 00 01 86 ca 02 f1 00 00 00 00 7f ff ff ff 00 00
0x40: 00 00 02 00 00 02 00 00 00 0c 00 00 66 66 68 68
0x50: 78 7a 82 66 00 00 00 10 47 11 20 48 8a 08 45 00
0x60: 1f 00 3f 43 45 f1 f3 f5 f7 f9 fb fd a8 ff 20 f2
0x70: 02 1e 34 5c
  3.676 - Aperture control stopped (1.777 s)
  4.000 - 1pps + time msg 10003
  4.087 - TM transfer started
```

No 1PPS or clock message

```
  0.000 – start simulation
  1.680 - Aperture control stopped (- s)
  1.688 - memctl, switch EEPROM page (p0.0) to (p1.3)
  1.697 - memctl, switch EEPROM page (p1.3) to (p2.3)
  1.706 - memctl, switch EEPROM page (p2.3) to (p3.3)
  1.896 - Start Aperture control close
  3.159 - TM transfer started
  3.197 - itf transmitted (116 bytes)
0x00: 64 fa 30 04 16 00 6d 40 00 00 20 21 00 00 00 fe
0x10: 00 00 00 00 0c 82 c0 00 00 59 00 0f 42 40 20 c3
0x20: 01 00 00 00 00 00 00 ff ff fe 00 00 00 10 00 00
0x30: 00 01 86 ca 03 04 00 00 00 00 7f ff ff ff 00 00
0x40: 00 00 02 00 00 02 00 00 00 0c 00 00 66 66 68 68
0x50: 78 7a 82 66 00 00 00 10 47 11 00 48 8a 09 cd 00
0x60: 14 00 3f 41 43 f1 f3 f5 f7 f9 fb fd a8 ff 20 06
0x70: 01 33 13 34
  3.687 - Aperture control stopped (1.791 s)
  4.159 - TM transfer started
  4.197 - itf transmitted (116 bytes)
0x00: 64 fa 30 04 7b 00 6d c0 00 00 20 21 00 00 00 fe
0x10: 00 00 00 00 0c 82 c0 01 00 59 00 0f 42 41 20 c3
0x20: 01 00 00 00 00 00 00 ff ff fe 00 00 00 10 00 00
0x30: 00 01 86 ca 03 fe 00 00 00 00 7f ff ff ff 00 00
0x40: 00 00 02 00 00 02 00 00 00 0c 00 00 66 66 68 68
0x50: 78 7a 82 66 00 00 00 10 47 11 00 48 8a 0a fb 00
0x60: 19 00 3f 41 43 f1 f3 f5 f7 f9 fb fd a8 ff 20 00
0x70: 02 1e b2 92
  5.160140 - TM transfer started
  5.198099 - itf transmitted (116 bytes)
0x00: 64 fa 30 04 15 00 6d 40 00 00 20 21 00 00 00 fe
0x10: 00 00 00 00 0c 82 c0 02 00 59 00 0f 42 42 20 c3
0x20: 01 00 00 00 00 00 00 ff ff fe 00 00 00 10 00 00
0x30: 00 01 86 ca 04 f8 00 00 00 00 7f ff ff ff 00 00
```

Note Simulation still needs to be rerun for the LAMP case this is still old simulation data

4.2 Telemetry Generation

The instrument generates both low speed telemetry and high-speed telemetry. Both are transmitted via a redundant interface to the spacecraft. Data on primary and redundant interfaces is identical. The housekeeping telemetry generation starts a few seconds after power up and in nominal operations does not cease until the instrument power is switched off. The high-speed science telemetry is only generated when the instrument performs acquisition operations.

The data handling system of the spacecraft will further influence the telemetry timing. For the low speed data either all telemetry or a sub-sampled set may be forwarded immediately to the ground. In addition to this all telemetry will be recorded on the Solid State Recorder (SSR) from which all or sections may be later on selected for download.

4.2.1 Low Speed Telemetry Generation Timing

Note Simulation still needs to be rerun for the LAMP case this is still old simulation data

After receiving a 1PPS pulse, the housekeeping task starts gathering the required telemetry data. When the complete TM packet is build, the TM transfer will start and sending of small blocks of bytes to the TM FIFO will continue until the complete packet has been send. If the transfer could be sustained at a continuous transfer rate of 38k4 baud, the transfer of the TM packets would be completed after:

Nominal TM packet (ITF header + HK packet)	132 bytes	30 ms
Data dump TM packet (above + MD packet)	280 bytes	68 ms

In the system most tasks have a higher priority than the TM transfer task and, thus, the effective transfer time may be larger as the actual transfer will occur in interrupted chunks. Note that in case a data dump is being generated the software in parallel to the sending out of a larger TM packet also builds the next data dump packet so this operation also increases the software execution load.

Note Simulation still needs to be rerun for the LAMP case this is still old simulation data

On the actual instrument some measurements were performed that confirmed the TM timing for the nominal (idle) situation. In the software-simulated environment a simulation was run with various loads on the system the following values were found for the TM transfer link usage:

Table 5: Low Speed TM transfer timing

Activity	TM start delay	TM transfer duration	TM complete time
	FM hardware measurement: times in seconds		
idle	0.082	0.040	0.122
	Simulation results: average times in seconds		
idle	0.087	0.034	0.121
dump (memory)	0.147	0.180	0.327
idle	0.087	0.033	0.120
checksum	0.137	0.569	0.706
idle	0.088	0.033	0.121
testpattern	0.130	0.623	0.754
acquistion	0.088	0.033	0.121

Note Simulation still needs to be rerun for the LAMP case this is still old simulation data

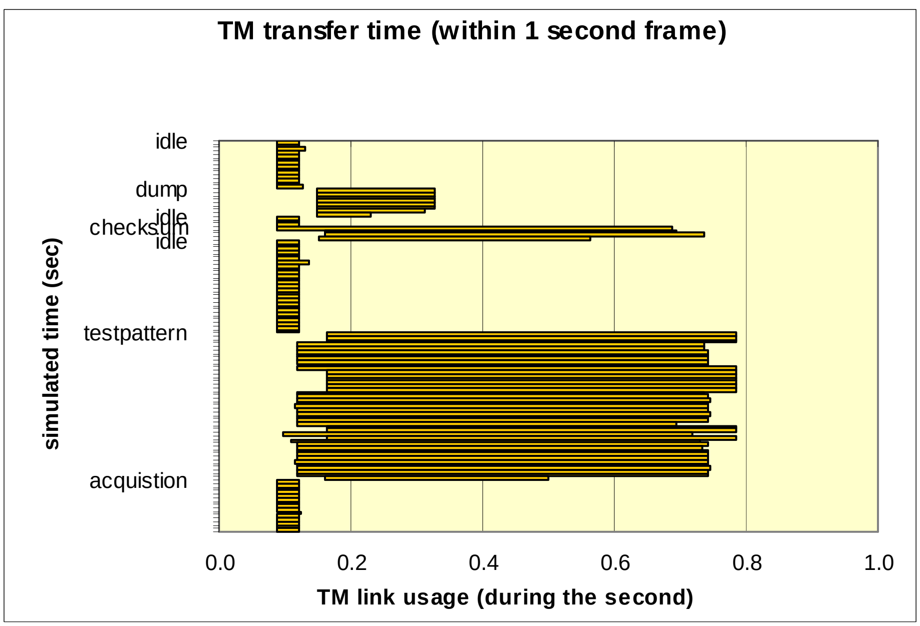

Figure 10: Simulation results: Low Speed TM transfer timing

In the dump phase a double effect is shown, both the needed TM transfer time increases and the load on the system increases as the gathering of data for the next dump packet proceeds. At the end of the test, a test pattern acquisition was started; initially this puts a very large load on the system during the generation of the test pattern. Once the pattern generation is complete though, the software has hardly anything left to do, and the situation is similar to an idle situation. This situation corresponds to a normal acquisition where the software after startup of the hardware acquisition has hardly anything left to do.

4.2.1.1 High Rate (Science) Telemetry

The high rate science TM transfers from the instrument to the spacecraft is almost completely controlled by instrument hardware; the software is only involved in controlling this interface indirectly during acquisition control. The software controls the start and stop of acquisition operations and in case of a histogram acquisition the exposure time. In pixellist acquisition mode, the rate at which the acquisition buffer is filled is determined by the observation and the selected time hack rate. Science data transfers will occur whenever and acquisition buffer is filled. The data transferred via this link from the instrument to the spacecraft always consist of complete frames of 32768 16-bit words. After storing this data in the SSR, the spacecraft will take case of sending this data to the ground for this transfer the science frames will be packaged into a large number of CCSDS transfer packets.

Transfer of a single 32 kword (16-bit) science frame via the serial LVDS interface to spacecraft data handling system takes 17 clock pulses for each 16-bit word to transfer. At the 1MHz clock rate this results in a complete frame transfer time of: (16+1) * 32768 / 1MHz = ~0.5 sec.

The CCSDS packets used by the spacecraft to transfer the science data to the ground can contain up to 480 bytes. This means that to transfer a complete (un-compressed) LAMP science frame 137 CCSDS packets are used.

4.3 Telecommanding

Telecommands are sent to the LAMP instrument in the data field of command message ITFs as described in Appendix B. Both telecommands and time synchronization messages are send via this interface in the ITF format to the instrument. In nominal operation, the spacecraft will every second send a time message to the instrument and up to one telecommand may be send every second. The section continues with some general description of the two formats and the processing of the two message types followed by a detailed description of the processing applied to the incoming data stream. This description is complex, as it both needs to deal with input from the redundant interfaces and needs to be capable of handling numerous error conditions.

4.3.1 S/C Time Message

Each second the spacecraft data handling system will send a Time Message to LAMP via the serial RS-422 interface. This Time Message will be transferred as an ITF formatted packet. This message will contain the S/C time that is valid at the next time synchronization pulse. The transfer of the time message will be completed not later than 50 milliseconds before the next time (sync) pulse. The message data for the S/C Time Message is shown in Table 22. The size of the S/C time message is fixed and the message length field in the ITF is always 5. Under off-nominal conditions, the S/C C&DH may stop sending the S/C Time messages.

The time message together with the periodic synchronization pulse is used to synchronize the instrument clock. In addition, the time message also contains the ‘memory dump’ allowed flag that determines whether the instrument is allowed to include memory dump data in the next generated telemetry transfer.

4.3.2 Instrument (Tele)command Message

The general structure of the LAMP telecommands is based on heritage from the New Horizon mission as shown in Table 23, but LAMP does not verify the 32-bit CM checksum field. The number of command words and the meaning of the values depend on the specific command as specified by the Op-code field. Two commands are special in that the spacecraft defines the op-code and structure; they are the memory load and memory dump telecommand. For LAMP, these commands are described / executed just like any of the other telecommands.

LAMP uses a compact command set with 25 different telecommands to control the operation of the complete instrument as listed in Table 29. Each second, LAMP is capable of handling (starting execution) of a single telecommand in addition to the time message. For commands for which the timing is critical, like the acquisition command, the actual operation will start at the next sync pulse (or assumed pulse if no real sync signal is available).

Except for the memory function related telecommands, all commands are small and require either no parameters or only a single word. So, the overall LAMP telecommand size for the nominal telecommands is either 8 or 12 bytes. The memory function telecommands though are not meant to be used during nominal (science) operations.

Telecommands that can be executed immediately and that take hardly any time will be executed directly by the task that receives the telecommand. Some telecommands take inherently more time to execute, for instance because the command generates multiple memory dump packets or includes extended hardware activation. In these cases, the telecommand receiving process verifies the parameters and activates the processing of the telecommand by a separate task. From this point on, the telecommand processing task is ready to receive (and process) a new telecommand.

4.3.3 Command Processing

The overall command processing is a fairly complex function as it both handles the redundant telecommand channels and it also deals with the error detection and recovery. Also included in this processing is the general command processing that is performed before the actual command is executed after being dispatched to the appropriate module.

The telecommand dispatch and processing is not dependant on the occurrence of the clock synchronization pulse (1PPS). Most commands don’t even require any form of synchronization and are executed immediately; only the acquisition start and stop are synchronized. This independence of the synchronization signal means also that the instrument in principle can handle multiple commands per second as long as the internal 256-byte software buffer doesn’t overflow and the command processing doesn’t take more time than available. This means that the instrument can easily handle multiple simple commands for which the execution is nearly instantaneous or even that a ‘start acquisition’ command together will the required ‘confirm critical’ command could be send to the instrument within a one second timeframe.

4.3.3.1 Redundant channel handling

The software is initially listening for the command input from either serial channel. As soon as a single byte is received from a channel, this channel becomes the active channel and the data from the other channel is discarded (see Figure 11). This initial condition may be overridden by a selection in the parameter file that disables a specific channel is case of known problems. Once a telecommand is received and handled, the input processing remains in the same state (listening to the selected active channel) whenever the S/C communication remains nominal, meaning that the synchronization pulse is received on the same side as the telecommand.

The detection of a command error may occur at various levels within the telecommand decoding function. Whenever the synchronization signal disappears on the active side or too many errors are detected in the received commands, the input processing restarts itself to the condition of listening to both sides. This selection process results in a smooth switchover between the primary and redundant telecommand channels without the need for an explicit reset of the instrument both in the presence (nominal case) and absence of the spacecraft synchronization pulse; an overview of the handling of the different nominal and non-nominal TC cases is shown in Table 6.

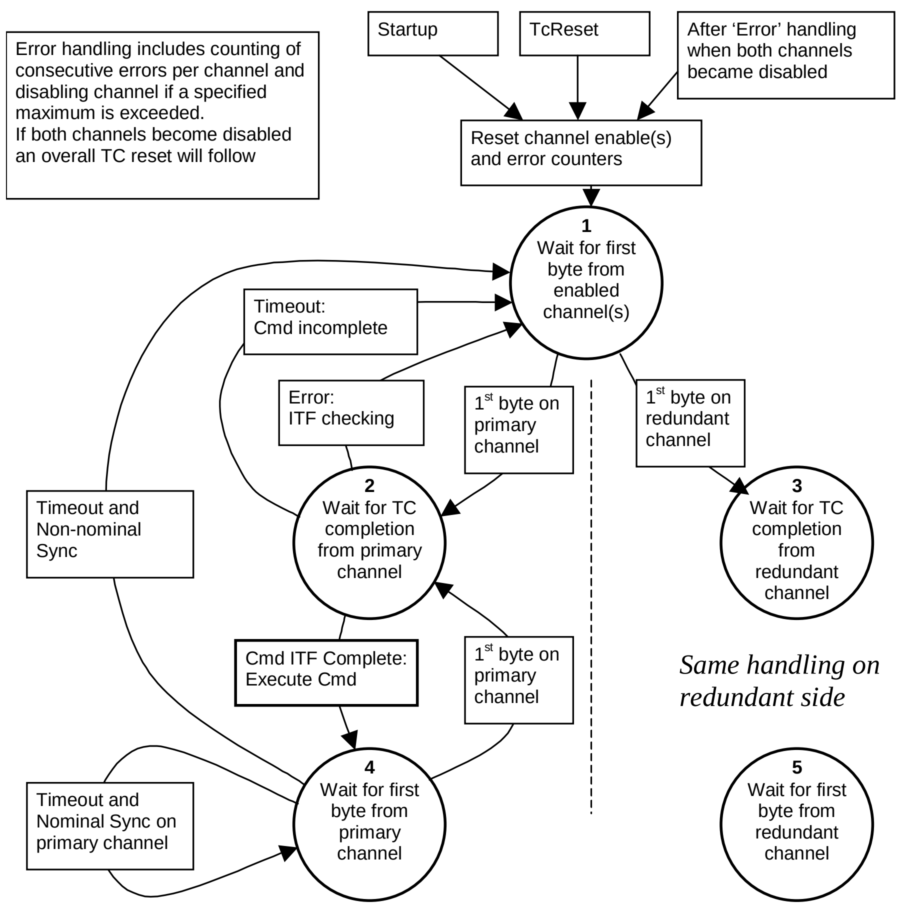

Figure 11: Primary - Secondary command processing

Table 6: TC processing behavior

Situation	Behavior	Switchover time
Nominal situation, TC and Sync active on a single channel	Telecommand acceptance follows active channel	Direct (next second)
No sync, TC on either channel	Telecommand acceptance follows active channel	Channel retention timeout (few seconds)
Commands on both channels	First detected byte determines active channel, remainder of data from inactive channel is discarded	Channel retention timeout (few seconds)
Sync only on other channel	Telecommand acceptance follows active channel	Channel retention timeout (few seconds)
Partial command	After command timeout the system returns to the waiting for initial by state	Not applicable

4.3.3.2 Character (Byte) input processing

Telecommand processing starts with data bytes being received by either one of the hardware receive UARTs (see Figure 12). These UARTs includes a 20-byte internal hardware FIFO buffer. This buffer is smaller than the maximum size telecommands (memory load telecommands can be up to 144 bytes + 7 bytes ITF header in size). The software (ISR TMTC) provides an additional buffer function that decouples the command processing from the input stream. This software task (indicated by a circle in the diagram) that handles the input from the UART only uses data from a single source at any given moment, determined by the current active side. This software is able to transfer the full rate input (transfers at 38k4 baud resulting in a maximum of 3840 bytes per second) into the internal software buffer.

The worst-case arrival time of the telecommand bytes at 38400 baud is once every 0.26 milliseconds; this means that the hardware FIFO can buffer little over 5 milliseconds. Within that period the software has to be able to transfer the received data to the software buffer. This second level software buffer is capable of storing 256 bytes corresponding to worst-case 66 milliseconds of data. During nominal operations this maximum should never be reached though, as only one telecommand and one synchronization message are to be handled every second and their maximum sizes are smaller than the available software buffer size.

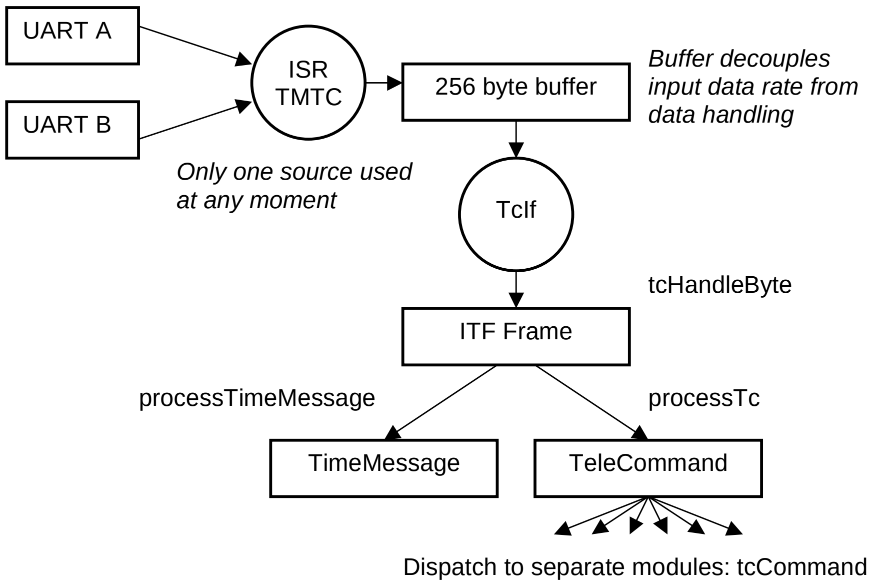

Figure 12: Command input handling

4.3.3.3 ITF reassembly and checking

The software (TcIf task as shown in Figure 12) reassembles the stream of received bytes into the ITF transfer frames using a state machine. This state machine decodes the synchronization pattern and the following ITF header bytes and then reads as many bytes as indicated. If the telecommand is not completed within a specified period, the decoding times out and the decoder returns to its initial state. Received ITF are checked before being accepted. The checking includes the following verification:

- ITF message type
- Telecommand (ITF) checksum

ITF message size

Once a complete ITF frame has been received the processing determines whether a time message or a command message was received. Received time messages are used for the instrument Clock function (see 4.4.3) where the received data (time and dump allowed status) is used for the time synchronization. Received telecommands are further checked and forwarded to the appropriate module for processing. Additional checking of the telecommand package includes:

ITF length versus indicated command length

Valid command opcode

- Indicated length versus expected command length

The command checksum is not included in this check for a correct command package. In case a problem is detected, the command decoding is aborted, an appropriate error message (see section 4.4.2 and Appendix G) is generated and included in the telemetry data and the system starts waiting for the next command.

4.3.3.4 Critical command processing

The LAMP instrument provides two levels of command verification to prevent erroneous commanding of conditions that might cause damage to the instrument. The first level of protection is provided by the operational state selection; not all commands are allowed in all states. In addition, a 'critical' command mechanism is used that requires confirmation for certain types of commands. These critical commands are different from the critical commands mentioned in the ground system; there the critical command system prevents sending command to the spacecraft until that operation is confirmed.

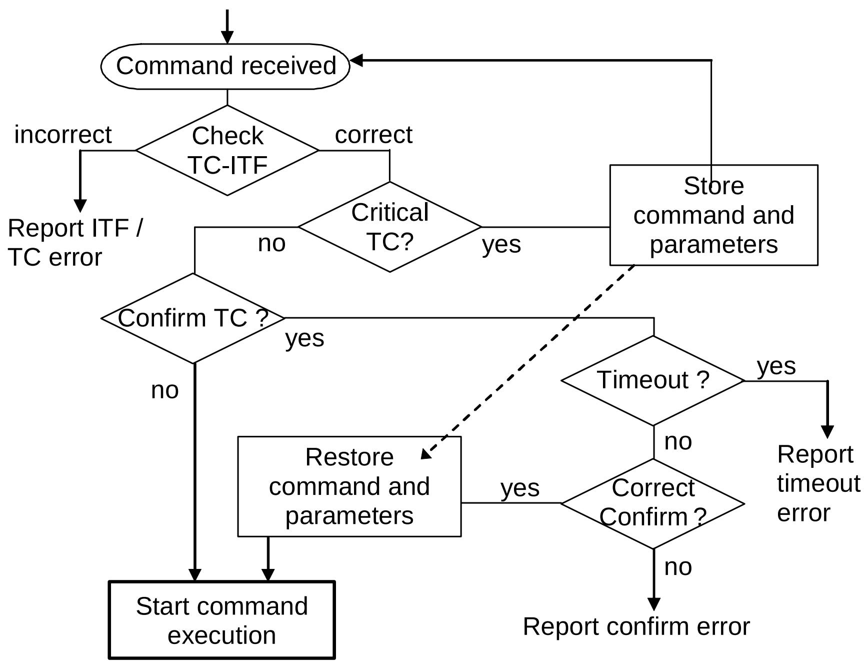

Figure 13: Critical command handling flowchart

Commands that are marked as critical such as ‘door opening’ and ‘high voltage activation’ are related to the safety of the instrument. These commands require a two-step commanding process before they are executed. After receiving such a critical command, execution doesn’t start immediately but the command is initially stored in an internal buffer. Once the confirmation command (a separate telecommand) is received within the timeout period specified in the parameter file (P_CMD_TIMEOUT), the command execution will start. If no confirmation is received within this timeout, a confirmation is received for the wrong command or another command is received, the initial command will be discarded. This failure will be indicated in the housekeeping packets. Figure 13 shows a flowchart to illustrate the basic functionality of the critical command handling.

4.4 General Instrument operations

4.4.1 Instrument state

In the previous chapter the instrument states were already described and a state transition diagram was presented (Figure 8). Here the states (Table 7) and the state transitions (see Table 8).are described in more detail.

Table 7: Operational states

State	TM	Power	Description
Off	N/A	Zero	Instrument not active
Safe	Only HK TM	Minimal	HV and heaters off
Acquire	HK TM and Science TM	Nominal (HV active), may include initial door opening/closing	HV On (but level may be set to zero) automated safety checking active, acquisition hardware active (heaters may remain at commanded setpoint)
Checkout	HK and Memory Dump TM (memory dump packets are only included when requested and when allowed by the s/c)	Varies, HV, door operations, actuator and/or heaters may be activated	Used during test, verification and commissioning, manual commanding, safety checking active (when HV activated), heaters may be commanded on

Table 8: State transition actions

Transition	Allowed	Actions
Safe to Checkout Safe to Acquire	Only when safety timeout is not active (or override is active)	Clear reported 'last active safety'
Checkout to Acquire	always	None (acquisition start is part of the command processing)
Checkout to Safe Acquire to Safe	always	- HVPS off, - stop actuator commanding, - stop heater control, - close aperture door (if enabled and actuator bus powered)
Acquire to Checkout	always	None, posting the synchronized acquisition stop is part of the command processing

4.4.2 Error reporting

The housekeeping packet includes three counters that indicate the acceptance and execution state of received telecommands. One of these two counters will be received whenever a telecommand is received. In addition, an executed command counter is reported that will be incremented whenever a command successfully completes execution. Whenever the rejected command counter is incremented, or when the executed command counter is not incremented after a command was accepted, an error would be reported. There are also some other cases where errors could be reported that are not directly related to execution of telecommands.

Two error-reporting mechanisms are implemented in the instrument. In each generated TM packet (and in the non-packetized telemetry) the most recent detected error and, if applicable, the related telecommand are reported. In addition to this mechanism a 16-entry circular buffer error log is maintained.

4.4.2.1 Most recent error reporting

The housekeeping packet contains the following fields:

- TcLastFailCode- last reported error code,
- TcLastFailed - last failed telecommand.

These two fields will be updated whenever a new error is detected. The last failed command code is a one byte field that contains the low order byte of the command code that caused the problem. In some cases, for instance, when errors are not directly related to telecommands being executed, or when no telecommand was yet recognized, only the error code filed will be updated. The error codes used by the instrument are listed in Appendix G.

The same error codes remain reported in the housekeeping packet until either another error is detected or when the instrument is reset. When multiple errors are detected in the same one-second period, only the last error code will be reported. After power on (or hardware reset) the last failed command will be set to 'no command' (special code ccNO_CMD = 0xff) and the error code will be set to indicate the power up reset state (special code ecNO_ERROR_INITIAL = 0xfe). Both error reporting fields are reset whenever the instrument receives the Tc_RESET_TC_STATUS command. In that case, the error code will indicate the reset state (special code ecNO_ERROR_RESET = 0xfd). This error reporting mechanism in addition to the three 'command' counters should be sufficient for 'nominal' error status reporting.

4.4.2.2 Circular error log

To support additional error investigation, the instrument implements a circular error log. The contents of the buffer may be dumped using the available memory dump command. For each detected error, this log contains time of occurrence, the error code and some status information. The error log uses the same basic error codes as reported in the housekeeping data. The error log provides a way to investigate in case multiple errors per second are reported or when not all housekeeping packets are available for investigation.

To keep the mechanism simple, the size of the circular buffer is limited to the size of one memory dump packet (128 bytes = 8 entries). The error log is located at the fixed address 0x8200 in normal memory data space (DATA).

At startup the circular error log is initialized with time values 0x46726565 ('Free') and no error code (ecNO_ERROR_INITIAL = 0xfe). The error pointer is initialized to the first entry. The error pointer is never reset during nominal operations. When a Tc_RESET_TC_STATUS is executed an additional special entry is added to the log (special code: ecNO_ERROR_RESET = 0xfd). In any case the next entry to be used in the error log is indicated by a special entry (ecNEXT_ERROR_LOG_SLOT = 0xf8).

4.4.3 Instrument Time

The instrument maintains a Spacecraft Time with a resolution of 1 second, but stores the received fractional time for reporting. The instrument does not maintain the time with a higher accuracy than one second, although a number of operations are synchronized to the received 1PPS pulse. In normal use, this time is synchronized to the spacecraft MET by means of the time message and the 1PPS synchronization pulse. The LAMP instrument was designed such that performance will be gracefully degraded when either or both of these mechanisms are unavailable. If the 1PPS pulse becomes unavailable, the instrument will start generating an internal (assumed) clock event that is derived from the X-tal oscillator and use this for all internal timing. The time message also transfers the memory dump allowed flag from spacecraft to the instrument. If no time synchronization message is received, the MET value will simply be incremented at the next 1PPS pulse (real or assumed).

The instrument concept of time consists of a 32-bit counter that counts the seconds since some given moment. A 32-bit integer number allows the specification of a period of 2^{32} = 4294967296 seconds (about 136 years); this should be enough for even an extended mission. In the nominal operation, a time update is received every second the time is received from the spacecraft in a time message; the value reported in this message becomes valid at the first following synchronization pulse. The module does not check the received time message for any specific value or range; any received time value is accepted and will become active at the next synchronization pulse. When no time message was received in the second before the synchronization pulse, the current time value will simply be incremented. This means that in case the time message generation ceases the instrument will still report incrementing (unique) time values. If no time message is received after an instrument reset, counting will restart from a fixed value (1000000).

4.4.3.1 Clock synchronization

The clock synchronization is a very simple software task that spends most of its time waiting for the synchronization signal (1PPS). If this signal is received, the instrument time will be updated; this means that either the time value received in the last second (time message) becomes active or that the time counter is simply incremented. In addition to this time update, the received 1PPS pulse is also used to synchronize the start and stop of acquisitions.

Whenever no synchronization signal is received after a waiting period (consisting of the 1-second nominal time and a 100 millisecond margin) the system 'assumes' a missing synchronization signal and the time update (and acquisition synchronization) is triggered in the same way. After this, the software starts using an internal generated synchronization signal that is generated every second to trigger the Clock module. This internal generation is terminated whenever a new external synchronization signal is detected. To prevent synchronization events occurring to close together (when transitioning from the internal to the external synchronization) the first received external synchronization pulse is discarded before the nominal synchronization processing is resumed. Note that there is no provision to recover from too many synchronization signals being received. This anomaly is considered too unlikely to validate the increased complexity needed to handle this.

4.4.4 Parameter storage

The instrument software includes a parameter file that stores a number of parameters that control instrument operations. For parameters that only occasionally change, this mechanism allows for small telecommands while still providing operational flexibility. To facilitate operations, this parameter file was implemented as a general mechanism that stores all of these constants in one location. The parameters in this table are listed in Appendix F. The contents of the parameter file are loaded into data memory from EEPROM memory whenever the system is started or restarted (see Figure 14) and these values in RAM are used during operations. At any time after that, individual entries in the parameter file can be altered by command (Tc_SET_PARAMETER). These changes to the table values only affect the values stored in this copied (active) table. A change to the table can be made permanent (persistent) by first changing the value in this table (Tc_SET_PARAMETER) and then commanding a parameter table store operation (Tc_STORE_PARAMETERS) that copies the current RAM table values into the three redundant EEPROM tables. At any time a reload of the data memory version of the table from the EEPROM memory can be commanded (Tc_LOAD_PARAMETERS), restoring the table to its power up default values.

The physical storage location of the redundant parameter memory is at the end of the last three of the four EEPROM code pages. At the end of each of these pages a small area of memory is reserved for this parameter storage. The three copies are thus spread throughout the instrument memory and are not all in one location. Whenever the parameters are loaded from the EEPROM memory into the data memory system start-up or explicitly by telecommand, see Figure 14), the values from all three tables are read and a three-way voting mechanism is used to determine the correct value. The following Table 9 shows how the system handles the different voting situations. Reporting of possible problems will use the command failure code as defined in the housekeeping packet (Hk_LAST_FAIL_CODE, see also Appendix G).

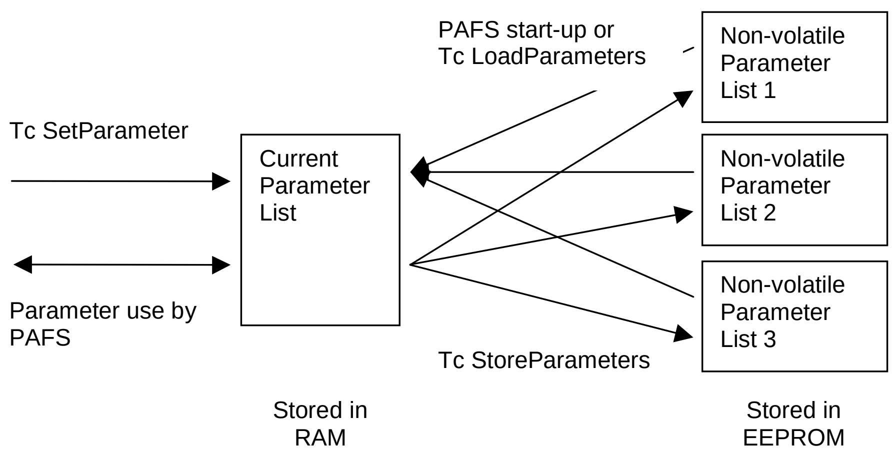

Figure 14: Parameter List storage and transfer

The 'permanent' version of the tables is stored in EEPROM memory. This memory can only be modified a limited number of times (hardware components are rated for 10000 page mode write cycles (see section 4.7.2), so it should be more than sufficient for the nominal mission operations). To keep track of the number of modifications made (write cycles), an additional count value (P_NumberOfModifications) is maintained in the table as the last entry. As part of the commanded table store operation (Tc_STORE_PARAMETERS), the current count value will be incremented so this entry always stores the number of performed write cycles. This telecommand can only be executed when the system is currently executing code from PROM. This is a hardware limitation, as otherwise the code execution will conflict with the EEPROM write operation (see section 4.7).

Table 9: Parameter File majority "Voting"

Voting Case	PAFS Action
All 3 agree	Use the value for the parameter
Two agree	Use the value of the two tables that agree for the parameter value, report the problem
All 3 different	Don't change the parameter value, go to SAFE state, report the error

The system implements a mechanism to verify the current contents of the parameter table. In order to limit the required telemetry bandwidth, a single (byte size) parameter is reported in each housekeeping telemetry packet (Hk_PARAM_VALUE). In order to identify the current reported parameter value; the index of this value is also included in the housekeeping telemetry packet (Hk_PARAM_INDEX). Reporting of a specific parameter can be requested by specifying the requested parameter by telecommand. This parameter is stored in an entry in the parameter list so the parameter to be requested is specified by the 'SetParameter' telecommand. This results in reporting the requested parameter starting with the next telemetry packet. This reporting continues until a different parameter is selected. To facilitate verification of all the parameter values (e.g. during testing) a special value is defined (cyclic reporting = 0xff) which will command the reporting of all the parameter values in successive housekeeping packages. In this mode, the identification of the current reported parameter value (Hk_PARAM_INDEX) cycles through all the parameters indices. Again, this reporting will continue cyclical until a specific parameter is requested. In addition to this the available memory dump command can always be used to get a snapshot of the complete contents. Either the current active table that is stored in DATA memory at address 0x8300 or one of the non-volatile tables stored at address 0x7f80 in the first three EEPROM pages can be requested.

The system also supports a directed read in which a parameter of the 'LoadParameters' telecommand specifies from which of the three tables the data should be loaded. In this case only a single table is read so no voting is applied. This operation is only intended for non-nominal operations where the system operator needs to circumvent known problems or needs to verify separate memory banks. In addition to this the table can also be initialized with 'hard-coded' values, this loads default values from PROM memory into the table; these values are the same values that are used to initialize the table before the initial 'start-up' load is performed.

4.4.5 Aperture Door Operations

After release of the launch latch, the airglow aperture door can be operated multiple times during the mission. A simple Limited Angle Torque (LAT) motor moves the aperture door. Two read-back sensors are used to determine positively if the door is fully open or fully closed. Control of the door uses a very simple open loop mechanism. To operate the door the control is always activated for the parameter file specified control period (P_DOOR_CONTROL, nominal 1.8 sec). No feedback from the reported door state to the door control function will be implemented (open loop control).

The door sensors use optical switches; in order to limit power consumption, the optical switches are only activated when a door measurement is needed. This same sensing system is also used for the detector door position switch. The system determines the position of both doors periodically once per second.

The two aperture door position sensors allow for positive verification of both the opened and the closed state. For the detector door (one time operation) only a single switch is available that can verify the opened position.

4.4.5.1 Aperture door performance measurement

In order to monitor wear of the door during the lifetime, a measurement function is implemented that measures the door operation times. This consists of a special function, which performs a detailed timing measurement of the door operation. The special function can be activated using the TcStartProgram telecommand specifying the start address of the function (0x30df). The special function will temporarily stop normal processing but since the performance measurement is completed so quick no telemetry data will be lost. After measurement completion the nominal instrument operation will continue.

A single activation of the measurement function will perform four time measurements with a 0.1 ms resolution. The first two measurements report the results of the open movement of the door by reporting start and completion of the movement. The start of the movement is defined as the moment the door leaves the first opto switch and the completion is defined as the time when the end opto switch is reached. The second two measurements perform the same on the closing operation. All four values are reported as 16 bit unsigned integers. The results of the measurements are stored in a 128-byte circular memory buffer at address 0x8500 in DATA memory. The memory dump function can be used to retrieve these measurement results.

During radiometric calibration the following values were measured under vacuum for the door operation times.

Table 10: Measured Aperture door movement times in vacuum

	Movement start	**End reached**	**Total Time**
Aperture door open	15.2 ms	59.2 ms	74.4 ms
Aperture door close	13.9 ms	60.1 ms	74.0 ms
Values based on measurements performed during the LAMP instrument TVac (2/1/2008), vacuum but not zero-G.			

4.4.6 HVPS control

The control of the High Voltage Power Supply (HVPS) is a main task of the instrument software. The instrument contains two redundant HVPSs that power the (single) detector. Each HVPS is controlled through a voltage level setting, an enable (on/off) control and an arm control. Each HVPS is powered from the corresponding Low Voltage Power Supply (LVPS) and a HVPS can only be activated when the corresponding LVPS and thus spacecraft instrument power bus is on (see Figure 24 in Appendix A -). Nominally, both HVPSs are controlled in parallel, but two flags in the parameter file can be used to disable either one in case of problems.

The HVPSs are controlled automatically by the software as needed to support science data acquisition. In the CHECKOUT state the HVPSs may also be commanded directly by telecommand (Tc_ACTIVATE_HVPS, Tc_DEACTIVATE_HVPS). In both cases the same rules govern the operation of the HVPSs. Both supplies have separate monitors for the MCP voltage (McpV), the strip current (StripI), and the anode voltage (AnodeV); these parameters are measured and monitored by the Safety task.

Operation of the HVPSs is related to the safety monitoring. The HVPS control controls the HVPS activation, and, during this activation, the safety monitoring will periodically (every 100 ms) verify the measured parameters. This safety monitoring is a separate activity that may in case of problems override the HVPS control. HVPS control will command the HVPS hardware according to the rules described in this section. The relations of the HVPS control with the safety checking are the following:

- First, in parallel the safety checking will continuously monitor the HVPS performance and take corrective actions that override the HVPS control Function
- Second, if a safety condition is in effect, then the HVPS operating procedures refuse to activate the HVPS as long as such condition is in effect (and not masked).

Regardless of whether the HVPS is activated automatically or by direct telecommand, the HVPS activation and de-activation complies with the rampup procedures.

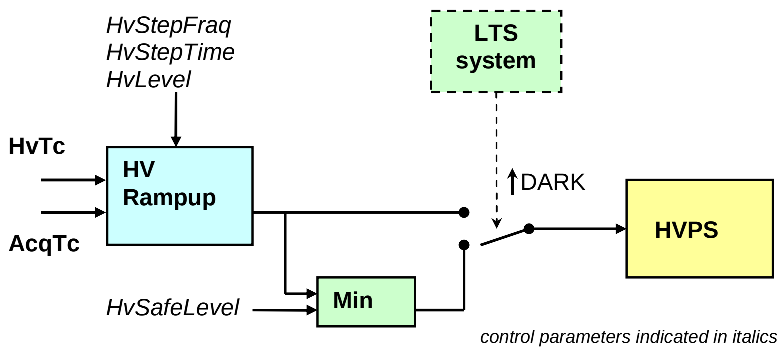

Figure 15: HVPS control signal paths

The actual HVPS output level may be influenced by the output of the LTS system. This system can when enabled (P_HvpsAuto) be used to limit the HVPS activation to a safe level (P_HvSafeLevel) at which the detector gain is virtually zero. This instrument automated system is planned to be used during the first phase of the mission to prevent the detector from being blinded by the sun lit side of the moon without direct ground (flight planning) input. This should limit the operational involvement of the LAMP instrument team to longer time planning. Later in phase II operations the same LTS signal is planned to be used to control the aperture door. Here the aperture door will be closed allowing observation of the sun lit side of the moon through the pin-hole that is build in the aperture door which will reduce the instrument sensitivity by about a factor of 500.

4.4.6.1 HV rampup

The HVPS control operates the redundant HVPS power supplies that provide both the adjustable Mcp Voltage and the fixed Anode Voltage to the detector. The turn-on/set procedure is used to set the HVPS to a programmable value. The HVPSs can only be turned on when the Instrument is NOT in the SAFE state. The HVPS set point value can either originate from the parameter file (P_HV_LEVEL) or from the HV on command (Tc_ACTIVATE_HVPS) parameter.

1. HVPS can only be commanded to any level that is lower or equal to the maximum HV level as specified in the parameter file (P_HV_MAX_HVSET), otherwise the turn-on procedure is rejected.

2. The turn-on procedure automatically enforces a (slow) ramp-up of the HVPS voltage level (not a sudden step):
   - If the commanded voltage level is above the current set-point (voltage will be increased) then the voltage setting is incremented by a step size determined by the step fraction parameter, with a step time (P_HV_STEP_TIME) between each increment. The step parameter may either specify a fractional or a linear stepping algorithm.
   - If the commanded voltage level is below the current set-point (voltage will be decreased), then the voltage setting shall be commanded immediately.
3. Turn-off procedure takes precedence over the turn-on procedure, either when manually started by means of telecommand (Tc_DEACTIVATE_HVPS) or because of triggering of the safety check.

A configuration parameter allows for the selection of either one or both HVPSs to be activated. The control provides a gradual ramp-up of the high voltage to allow for read-back monitoring and checking (by the Safety module) during the ramp-up to prevent damage to the HVPSs. Although hardware provides full redundant HV level control for both HVPSs, the system always commands both HVPSs to the same level. The ramp-up is controlled by two configuration parameters adjusting the step time (1-255 seconds) and the step fraction that determines the rate at which the step approaches the requested set point (see example in Figure 16). This mechanism results in a gradual decreasing steps when the commanded set point is approached. The mechanism is very flexible and can support slow ramp up in hours in the initial phases but also a quick operational ramp up in half a minute when sufficient confidence is acquired.

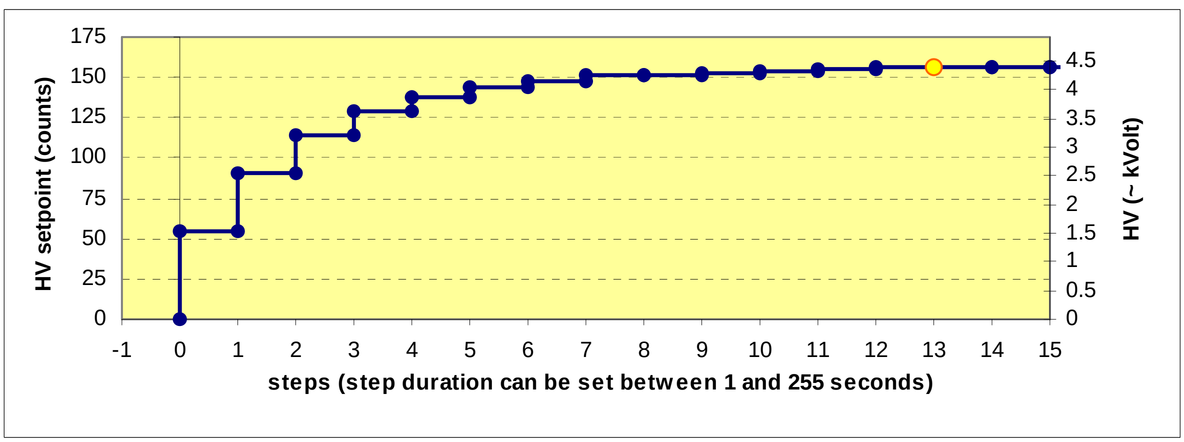

Figure 16: Example HV ramp-up from 0 to set point 157 (fraction = 45)

The algorithm below is used to determines the step size. The factor 16 in the step calculation is a constant pre-scaling of the HvFrac parameter used to obtain a useful adjustable range. In addition to this fractional control, the software also supports a linear ramp-up function; this mode is selected by specifying a value for HvFrac less than 16.

```
if (HvReq>HvSet) {          /* stepped increase   */
    if (HvFrac<16) {        /* -linear rampup     */
        step= HvFrac;
    } else {                /* -fractional rampup */
```

```
            step= [(HvReq-HvSet)*(16/HvFrac)];
            if (step == 0) step= 1;
        }
        HvSet= HvSet + step;
    } else {                        /* immediate decrease */
        HvSet= HvReq;
    }
```

4.4.6.2 LTS based HVPS limiting and Door control

The LAMP instrument is equipped with two separate LTS sensors; one is aimed a little ahead of the detector foot-print and the other one a little behind (when scanning the lunar surface). The dual sensor system provides redundancy but also allows for a more precise acquisition control based on a combination of these two control inputs. As the spacecraft will periodically change flight direction the role of the two sensors also changes. The software supports these operations by providing a very flexible control system controlled by a number of parameters. This includes an option to disable the complete operation of the LTS based safing mechanism. The LTS sensors are sampled about every 100 ms and each of these pairs of samples are used to update the state of the LTS-Dark signal that is used in the HVPS and door control algorithm.

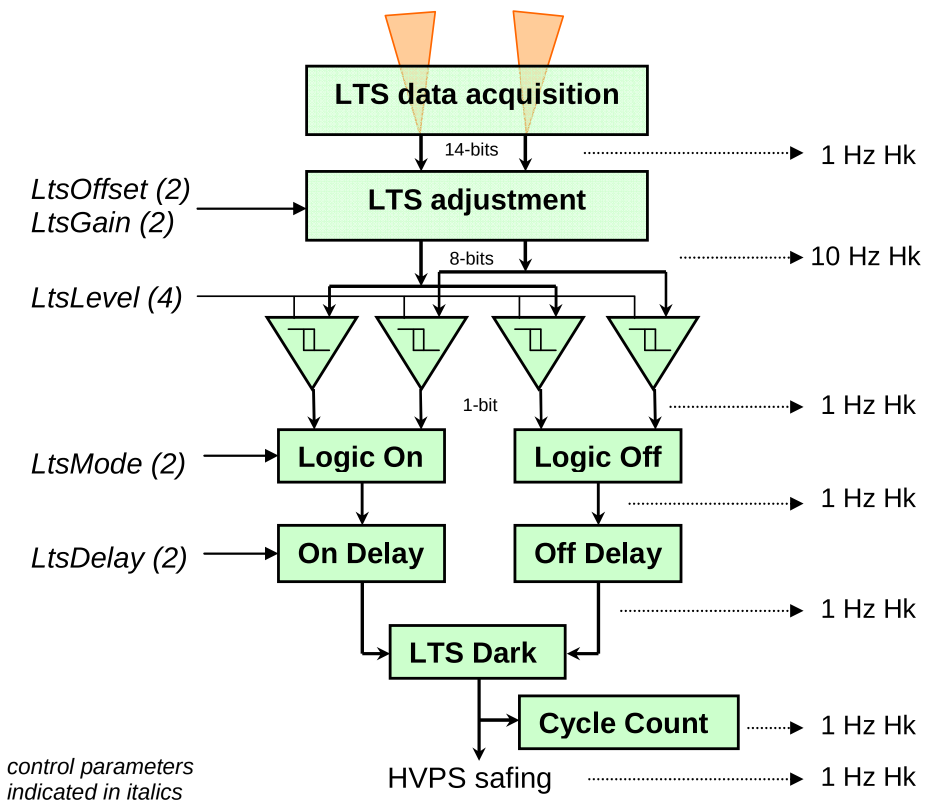

Figure 17: LTS signal processing and HK generation

The LTS system samples both LTS sensors every 100 ms and uses each of these analog values to determine the 'DARK' status. The overview of the processing is shown in Figure 17. When active this signal allows HV activation to the full commanded level, otherwise activation is limited to the safe level. The processing of the analog LTS signals resulting in the 'DARK' signal is as follows:

1. The analog LTS signals are sampled every 100 ms. Only the most recent Raw LTS value is reported each second in housekeeping data.
2. Both Raw LTS signals are 'adjusted' using a simple gain/offset correction, this allows for a software controlled selection of the most optimal dynamic range of the sensors:

   Lts_Adjusted = (Lts_Raw – Lts_Offset) >> Lts_Gain

   The resulting adjusted values are represented in an 8 bit value, up to 10 of these adjusted values for each sensor will be reported in the housekeeping telemetry.
3. Each adjusted LTS signal sample is processed as follows resulting in a possible update of the 'DARK' signal every 100 ms:
   - The analog adjusted LTS values (now range 0 to 255) are converted into a binary values based on simple programmable thresholds, separate thresholds for each of the LTS sensors and for the on and off functions are specified
   - The four binary LTS signals are pair wise combined using two separate selectable logic functions into a 'on' and 'off' signal. The results of the comparison to the high limits (greater then) are used for the 'On' signal and the results of the comparison to the low limits (less then) are used for the 'Off' signal. The two selectable logic functions allow for the selection of any possible logic combination of the two input variables (e.g. A-only, B-only, A and B, A and not B, always high, etc., see below for the specific functions)
   - The 'on' and 'off' signals control the activation/deactivation of the 'DARK' signal, based on separate programmable delays, these are implemented such that the state transition of the 'DARK' signal only occurs when the specified number of consecutive 'on' or 'off' signals is reached.
   - Finally the on and off signals are combined in a single 'DARK' signal, 'On' sets the Dark signal and 'Off' turns the Dark signal off, 'Off' has priority, so when both 'On' and 'Off' are active then the Dark signal is turned off. When neither is active then the Dark signal stays in the same state.
4. The state of the Dark signal determines the LTS control, based on the LTS control parameters (P_DoorAuto and P_HvpsAuto). If HVPS control is selected then the HVPS output voltage will be limited to the configured safe level (P_HvSafeLevel) unless the LTS indicates a Dark signal. This control takes place every 100 ms cycle, so HV control is active instantaneous. If Door control is selected then the door will be closed when the Dark signal is cleared and the door will be opened when the Dark signal becomes set. Note that this only result in an action at the transition so at the start of an acquisition the door will have to be commanded in the correct state.
5. The number of LTS based HVPS or Door control cycles (LTS-Dark transitions to zero) is counted per acquisition; this number is be used to limit the number of HVPS or door activation

cycles per acquisition. This number of cycles is also reported in housekeeping telemetry every second. At the start of an acquisition this counter is reset to zero.

The selectable logic functions for the LTS control allow for the selection of any function of two (boolean) input variables (see Table 11). The two input variables are the comparison to the high limits for the 'On' commanding and the comparison to the low limits for the 'Off' commanding. The two boolean input variables function as an index in the four bits table and thus point to the result of the function evaluation.

Table 11: LTS control functions (Mode)

Function value	Binary value	Function	Note
0	0000	Off (always)	Never active
1	0001	!(A+B)	
2	0010	A·!B	
3	0011	!B	
4	0100	!A·B	
5	0101	!A	
6	0110	AxB	Only one active
7	0111	!(A·B)	
8	1000	A·B	**Both active, default for 'On'**
9	1001	!(AxB)	Both the same
10	1010	A	A only
11	1011	A+!B	
12	1100	B	B only
13	1101	!A+B	
14	1110	A+B	**Either one active, default for 'Off'**
15	1111	On (always)	Always active

4.4.7 Actuator firing

The PERSI-LAMP instrument software controls the activation of all one-time actuators that are incorporated in the instrument. The controlled actuators consist of one redundant Wax Pellet Actuator (WPA) and three redundant Shape Metal Alloy (SMA) actuators. One of the redundant SMA actuator channels though is used to control the detector release valve. This detector release valve control uses the same control mechanism as the SMA although the used activation timing will probably be different. Nominal control allows for activating either the primary or the redundant side of each actuator, although for contingency operations both sides of a single actuator may be activated.

The activation (firing) of the actuator and the selection of the activated side are directly controlled by a telecommand, but the actuators can only be activated when at the same time the appropriate actuator (heater) power circuit is activated by the spacecraft. Independent from the software an actuator safe plug is used to disable the actuator activation during ground testing. Activation of any of the actuators may be terminated by a separate telecommand option. The SMA and WPA have very different timing and control requirements; control functions for WPA and SMA will be described separately in the next two paragraphs.

The LAMP instrument has four different one-time actuator mechanisms; partly these mechanisms are realized in a redundant way. The five actuator mechanisms operate the various protection mechanisms in the instrument:

- Detector door: one (redundant) WPA to open the door (once),
- Failsafe door: one (redundant) SMA to open the failsafe door (once, but not planned for the nominal mission),
- Aperture door latch: one (redundant) SMA to unlatch the door (a separate actuator operates the door after unlatching see 4.4.5),
- Vent valve one actuator (solenoid) to open the valve and one to close the detector vent valve (multiple operations, but not planned for the nominal mission).

4.4.7.1 SMA

The SOC door, failsafe door and aperture door latch are controlled by SMA actuators. For the SOC and aperture latch redundant mechanisms are available, the failsafe door has only a single actuator mechanism. The control of the SMA is directly time controlled; the control period is determined by a single shared configuration parameter (P_TINI_CONTROL), but during operations different values may be configured. This configuration parameter allows for a range of 5 to 1275 milliseconds of control (with a resolutionof 5 ms). Once activated the control remains activated for the defined control period. The required SMA activation time depends on the temperature (as shown in Figure 18), at room temperature a value of about 70 ms was used.

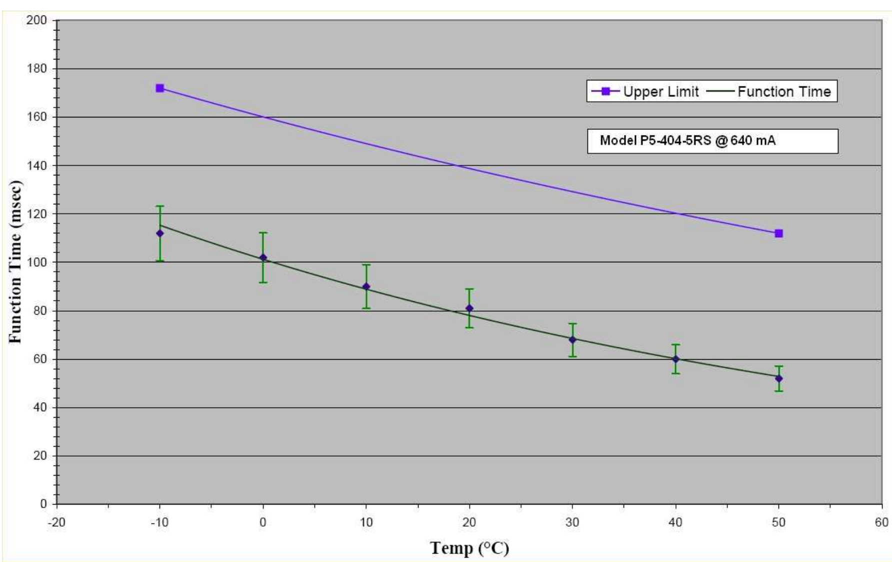

Figure 18: SMA activation times

4.4.7.2 Vent valve actuator

The detector housing vent valve actuator consists of a bi-stable solenoid that can be commanded open and close under software control. This vent valve can be used as a backup evacuation of the detector housing, but is not planned to be used during the nominal mission. This solenoid requires a fixed activation period just like the SMA actuators and the software treats is just like another SMA. For the detector release valve an activation time of about 5 ms was determined during testing to give a reliable operation.

4.4.7.3 WPA

The detector door actuator is a High Output Paraffin (HOP) actuator, also called Wax Pellet Actuator (WPA). In flight, the detector door can only be opened, not closed (during ground testing prior to launch, it can be manually re-closed). A wax pellet actuator with redundant bridge wires actuates the detector door opening mechanism. The detector door release mechanism has a sensor (switch) to indicate when the wax pellet actuator plunger has extended to a certain position and a separate sensor to indicate when the detector door is open. The software controls both bridge wires in the actuator and reads the status of both sensors. The WPA activation duration depends again on the environment temperature; in vacuum at room temperatures activation times of about 2 minutes have been observed.

The control of the WPA is also time limited but the activation period will also be terminated when the WPA feedback switch indicates that the active position is reached. This redundant scheme is used to prevent over-activating of the wax pellet actuator (and possible contamination). Note that the detector door position indicator is NOT used in the WPA control algorithm. The configuration parameter (P_WPA_TIMEOUT) in this case defines a timeout period after which the actuator is in any case switched off. The configuration parameter allows for a timeout period in the range of 10 to 2550 seconds (with a resolution of 10 seconds). A separate configuration parameter (P_WpaSensorEnable) allows for the disabling of the feedback switch; in that case, the control defaults to a purely time based control.

4.4.8 Heater control and temperature measurements

The instrument contains two sets of redundant decontamination heater on the OAP mirror and grating (four heaters in total). These heaters are used to raise the temperature of these two optical surfaces to clean them from contaminations. A control algorithm based on the temperatures measured on these optical surfaces is used to prevent overheating.

In nominal operations, this decontamination will be performed separate from any acquisition operations, although the software doesn’t prevent heater activation concurrent with acquisition operations. But heater commanding (TC acceptance) can only occur when the instrument is in the checkout state.

The control of the heaters uses a very simple on-off control to prevent overheating; both OAP mirror and grating have their own control. Whenever the last measured temperature is below the set point, the selected heater(s) of the corresponding redundant heater pair will be activated. For this control, each of the heated surfaces is equipped with redundant temperature sensors. A configuration parameter determines whether the primary or the redundant sensor is used for the control algorithm (P_HtrSenseGrating and P_HtrSenseMirror).

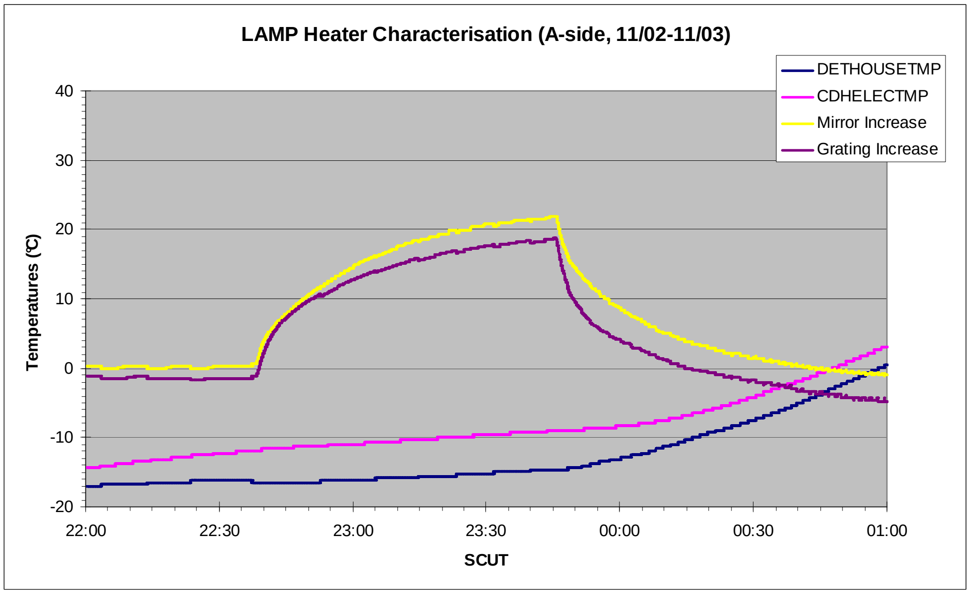

Figure 19: LAMP heater activation in vacuum, single side heaters

To prevent excessive switching, the control algorithm is only activated every 8 measurements, so, at a maximum, one switching action (on or off) per heater can occur per 8 seconds. The control of both redundant heater pairs is staggered in time so the switching actions of both heater pairs can never occur at the same time. The selection (and activation) of either primary or redundant heaters (or both) for each surface is again based on configuration parameters (P_MirrorHtr1Enable, P_MirrorHtr2Enable, P_GratingHtr1Enable and P_GratingHtr2Enable), but also the activation of primary and/or redundant actuator but may be used for selection.

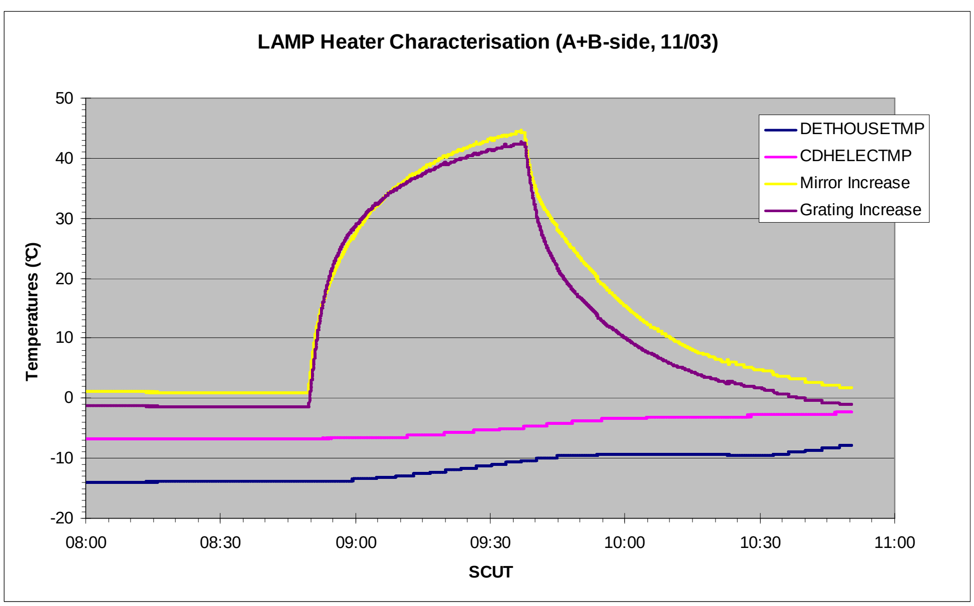

Figure 20: LAMP heater activation in vacuum, dual side heaters

In a vacuum environment the temperature responses of the optical surfaces were measured and even though no thermal equilibrium was reached the general trends are clear. The first activation only activated the primary heaters on all optical surfaces; in the second test both the primary and redundant heaters on the two optical surfaces were activated.

4.5 Acquisition processing

After commanding an acquisition start, the instrument will first go through an acquisition startup sequence that brings the instrument in the proper acquisition state. Most of the acquisition parameters are controlled by settings retrieved from the parameter file but two are defined by parameters of the acquisition start command. These startup actions were already presented in Table **2**. If the selected acquisition state was already commanded, this setup will hardly take any time but when door operations of high voltage rampup is involved the startup may take a considerable (deterministic) amount of time. During this startup the safety checking ensures that the high voltage activation does not endanger the instrument through monitoring of the safety parameters.

Next the actual acquisition is started; this can either be a pixellist or a histogram acquisition. A histogram acquisition may consist of multiple exposures of a duration defined in the parameter file that each result in the production of a separate science data frame. A pixellist acquisition stores all detected photon events together with time hack data. The time it takes for a pixellist acquisition to fill an acquisition buffer depends on the selected time hack rate and the number of photon events detected. The number of photon events detected depends on the brightness of the object and the detector area that is not masked with a hot pixel mask. Each time when an acquisition buffer is filled, a science data frame will be produced. Nominally the acquisition is terminated by a separate 'stop acquisition' telecommand. To prevent endless acquisition, the instrument implements an acquisition timeout function that terminates an acquisition automatically when the defined timeout period is expired. This timeout mechanism may also be used as the sole purpose of acquisition duration control. During acquisition the safety checking continuously ensures that the high voltage activation and does not endanger the instrument through monitoring of the safety parameters.

When an acquisition is terminated either by commanding the instrument directly to the SAFE state or by a safety event during the acquisition processing, the instrument will be brought in a safe condition. The HVPS will be disabled, heaters deactivated (if they were on) and the aperture door will be closed (if enabled and if one of the actuator busses was activated). When an acquisition is terminated by commanding the instrument to the CHECKOUT state or when the acquisition timeout was reached, the instrument remains in an active state and a new acquisition can be started immediately.

In addition to normal actuations the instrument also supports 'test pattern' acquisitions. These modes are available for interface verification. In these modes basically the normal acquisition sequence occurs but instead of producing actual acquired science data frames, pre-defined test patterns are generated and send to the spacecraft.

4.5.1 Acquisition Startup

The acquisition sequence starts after receiving the confirmation to an acquisition command. The startup sequence and the duration of the various steps are listed in Table 2. The acquisition start sequence can only start when the instrument is able to change to 'ACQUIRE' state, meaning that no safety condition is currently in effect (safety timeout equals zero). The start of an acquisition can either be commanded from the SAFE or from the CHECKOUT state. The difference between the two is that in the CHECKOUT state, the HVPS and/or the heaters could already be commanded on. This would for instance allow for a pre-verification of the stability of the MCP and anode voltages or an acquisition with heaters activated (used during decontamination operations). As a first step the acquisition hardware is configured for the acquisition by loading the acquisition (hardware) control registers with parameters from the parameter file (P_DISCRIMINATOR, P_StimEnable, P_PIXELLIST_HACK, P_HOTSEG_1-8).

The 'dooropen' parameter of the start acquisition command defines the commanded state of the aperture door. If the aperture door is already in the requested state (according to the door sensors), the commanding of the door will still take place but the startup sequence continues immediately. If the door was not in the requested state the door commanding will also take place but the startup sequence will pause until the door commanding is completed. Note that the door commanding period is a fixed period that is defined by a parameter in the parameter file (currently 1.8 sec). In this case the position reported by the door sensors is not taken into account anymore, the acquisition startup will continue after the defined commanding time irrespective of the (sensor) reported door position. These actions ensure that the door movements have completed before any acquisition activity is started. Note that the aperture door can only be moved when one of the actuator busses is activated. For the nominal histogram airglow observations, the Aperture door will be commanded open. For the pixel list occultation observations, the Aperture door may remain closed to prevent contamination of the occultation data by light entering the airglow aperture.

The next step is the high voltage activation. The HVPS activation always follows the rampup sequence as described in section 4.4.6. The acquisition startup only continues when the specified HVPS setting (as defined in the parameter file) is reached. This means that this step can be immediate when the HVPS was already commanded to the desired level, but the rampup may take a considerable period as defined in the HVPS rampup parameters in the parameter file. If during the HVPS rampup, a safety condition occurs the complete acquisition is terminated and the instrument returns to the safe state.

4.5.2 Acquisition processing

After these steps, the instrument is ready to start the acquisition. In order to start the acquisition at a deterministic point in time, the actual start of the acquisition processing is synchronized to the synchronization pulse (1PPS). The actual control of the acquisition hardware occurs about 2 ms after the instrument receives. At this point either the pixellist or the histogram acquisition mode is selected (hardware) and detector events are stored in the acquisition memory. For the pixellist acquisition, the acquisition memory actually fills; the detector events and time hacks are stored in a sequential list that fills the acquisition memory. The progress of this can be observed in the housekeeping data in the pixellist counter. For the histogram acquisition, the detector events are counted based on the spectral/spatial and amplitude 'position' of the event (histogramming). Then after a defined exposure time, the acquired image is transferred to the spacecraft.

In both cases, the instrument can perform continuous uninterrupted acquisitions. Meaning that there is no dead time in which the detector events are lost in between filling and switchover of an acquisition buffer. When the first buffer is filled, or an exposure is completed, acquisition continues in the second buffer. In parallel the first buffer is transferred to the spacecraft. The only case in which data could be lost is when the acquisition buffers fills faster than the data can be sent to the spacecraft.

In parallel to the acquisition operation by the instrument, the normal generation of housekeeping data continues providing ancillary data describing the conditions of the acquisition. One of the parameters reported in the housekeeping data is the measured raw countrate. This countrate value provides a temporal registration of the overall detected eventrate. This can be especially important for the histogram acquisition mode, as this mode normally does not include any temporal information.

4.5.2.1 Pixel List Mode

The pixel list mode is completely hardware controlled; after setup no further software actions are required, until the acquisition terminated by the software. The software determines the overall acquisition duration. Two acquisition buffers are used so acquisition in one buffer can continue while the contents of other buffer is transferred to the spacecraft. Switchover between the two acquisition buffers when a buffer is filled with pixel list data will be handled automatically by hardware; software will be informed when a buffer switchover occurs. At the end of the acquisitions, software terminates the acquisition mode, which will force transfer of the remaining buffer contents to the s/c (flush). Whenever the instrument generates a complete science frame, this will be reported in the housekeeping data; both the frame number of the last generated frame and the time (MET) at which the frame was generated are included in the housekeeping data. Note though that the generation of the science frame is data driven and that the acquisition buffer swap will not be synchronized to the spacecraft time. The reported frame time will be the current time, meaning the time (MET) that became current during the last time synchronization (1PPS).

During the acquisition, 16-bit words are written into the acquisition memory. The first word of the acquisition memory stores the frame identifier; so 32767 words are left to store the event and time hack data. Normal events in the pixel list memory are indicated by bit 15 of the word set to zero, the other 15 bits contain the spectral and spatial information of the event. Time hacks have bit 15 of the word set and the remaining 15 bits will be used to include a counter so timing correlation remains possible even when some data is missing (15 bits hack rate allows for 128 seconds of time information before the time hack counter wraps around). The most recent value of the time hack counter together with the science packet block counter will be included in the housekeeping data to allow time correlation.

The time it takes for a buffer to be filled with pixel list data depends on the actual count rate and the selected time hack rate. At the predicted nominal event rate for the dark side of the moon of about 1500 Hz and the highest time hack period of 4 ms, the 32-kword (32767 entries) pixel list memory could fill in 19 seconds. Close to the terminator with some bright objects coming into view the frame rate could increase to about one frame per two seconds. On the day side of the moon two methods will be used to protect the detector from too high countrates. Initially (firs month of operation) the HVPS would be lowered to a safe level so no photon events would be detected, but the timehacks would fill the acquisition buffer in 122 seconds. In a later operating phase when a basic data set has been captured, the aperture door will be closed over the day side of the moon and observation will be continued through the pin-hole, this results in a much lower overall sensitivity and the frame buffer would on average fill in 33 seconds. The transfer of a complete high-speed frame to the spacecraft will always take 0.55 sec (see section 4.2.1.1). If the current transfer is not completed before a second acquisition buffer is filled the instrument temporarily can’t store (acquire) and some data will be lost. This case though will only happen during abnormally high event rates (above 58571 photon events per second), well outside normal operational range of the instrument.

The instrument generates frames on the high-speed data link when the acquisition memory is full and, hence, the transfer is inherent a burst transfer. Still an average datarate can be calculated. Based on the three above-mentioned cases, this results in average datarates of respectively 28 kbps, 262 kbps, 4 kbps and 16 kbps.

As mentioned before, it is possible that due to a very high event rate (>58571 events/second) the science transfers to the spacecraft can't keep up with acquisition and data will be lost. This can only occur in Pixel List mode, as here the frame generation rate is dependant on the observed event rate. In these cases, an interruption in the acquisition may occur. This interruption is not a reason to terminate the acquisition but it will be reported in the HK data stream (Hk_HSTM_OVFLW_ST)**.**

Nominal science data acquisition is planned to be performed in pixellist mode with a timehack of 4 ms to obtain the required temporal (and therewith ground coverage) resolution. In the nominal 50 km orbit the spacecraft and the LAMP instrument will every 4 ms cover 6 meter of Lunar surface in the movement (perpendicular) direction. Table **12** lists the expected data volumes for the various pixellist acquisitions.

Table 12: Expected pixellist data rates

	Average photon rate	**Hack rate**	**Average Frame time**	**Data volume per orbit**
Night side operation; aperture door open	1500 Hz	250 Hz	19 sec	11.9 Mbyte
Day side operation; aperture door open (HV reduced)	18 Hz	250 Hz	122 sec	1.8 Mbyte
Day side operation; aperture door closed (pinhole observation)	750 Hz	250 Hz	33 sec	6.8 Mbyte

Notes:

- The predicted photon rates assume a nominal operating instrument with nominal sensitivity
- The numbers listed are based on typical estimates, 50 – 100 % variations in the sky Lyman alpha intensity are observed depending on solar activity
- The predicted photon rates include the STIM pulses (~18 Hz)
- These numbers originate from the sensitivity analysis that was described in memorandum …...

4.5.2.2 Histogram Mode

The instrument software controls the histogram mode exposure timing. The hardware performs the histogram operation in which events are counted. The raw pixel events will be histogrammed (counted) into the acquisition memory based on the spectral and spatial 'location'. The histogramming uses a read-increment-write operation to the addressed histogram memory (bin) location that corresponds to the spectral and spatial 'location'. So for each spectral-spatial 'location'; the number of events that were detected at that 'location' are counted. The increment operation will saturate at maximum count number, so when more than 65535 events are detected for a given location, the maximum number of counts will be reported. In addition, a second histogram operation will be performed on the event pulse height (amplitude) data. This results in what is called the pulse height data, a distribution of the amplitude of the detected events. This data will be stored in an otherwise unused area of the acquisition memory (beginning of second and third row of the array). This uses the same read-increment-write operation; this also has to use 16 bit 'event' counts (just like the main event histogramming); LAMP uses 6 bits to represent the pulse height resulting in 64 pulse-height bins.

The histogram acquisition does not ‘fill’ the acquisition memory like the pixellist acquisition does. The counting of events could continue and even when some locations (pixels) are saturated there is no hard reason to terminate the acquisition operation. To provide a temporal resolution of the acquisition a single histogram acquisition may consist of multiple exposures. When an exposure is completed the instrument switches acquisition buffers and the acquisition continues in a fresh (cleared) acquisition buffer. The duration of histogram exposures is defined by a parameter file entry (P_HISTO_EXP_DUR), and can be selected anywhere between 1 second to 65535 seconds (~18 hours).

The selected exposure duration defines when science data frames are sent to the spacecraft; in all cases the frames are again 32768 words in size. The size of the data frames is independent of the number of detected events, as a frame always represents the full histogram covering the detector range of 1024 spectral lines * 32 spatial columns. Data transfers again occur in bursts, the 32 kword frame is transferred in about 0.55 sec (see section 4.2.1.1). An average high-speed science datarate can be calculated that depends on the selected exposure rate:

$$\textit{average data rate} \quad = \quad \frac{524288}{\textit{exposure duration (in seconds)}}\ \textit{(bps)}$$

So for reasonable exposure duration of 100 seconds the average data rate for the high-speed science data would only be 524 bps.

Again each time an acquisition buffer swap occurs, the value of the Last Acquisition Completion time in the HK telemetry will be updated; this provides information to perform post acquisition timing correlation. Note that in case of a histogram acquisition the exposure timing is synchronized to the 1PPS pulse.

This will also happen when the acquisition is terminated as described below. This provides a means to track the acquisition timing relative to the spacecraft elapsed time.

4.5.3 Acquisition termination

There are three reasons why an ongoing acquisition operation may be terminated:

- Commanded ‘stop acquisition’,
- Acquisition timeout,
- Safety event.

Whenever the acquisition termination occurs synchronized the actual hardware acquisition process will be terminated about 2 ms after the instrument receives the synchronization pulse (1PPS).

Nominally an acquisition will be terminated by an explicit command that commands the system out of the acquisition mode and thereby terminates the acquisition operation. The acquisition termination will, just like the acquisition start, be synchronized to the spacecraft synchronization signal (next 1 second pulse (real or assumed, see section 4.4.3.1)), so the exact time of acquisition completion is known. To stop an acquisition the instrument may be commanded either to the SAFE or CHECKOUT state. When switching to the SAFE state terminates an acquisition, this will also result in the deactivation of the HV and when enabled (P_DoorEnable), the closing of the Aperture door (when one of the actuator power busses was active). When commanding the instrument to the CHECKOUT state terminates the acquisition, no further actions are taken.

In addition to this (direct commanded), the instrument implements a timeout mechanism that will terminate an acquisition after a defined period of time (P_ACQ_TIMEOUT). This timeout prevents the instrument from generating an unbounded amount of science data in case the 'stop acquisition' command is missed. When the acquisition is terminated due to an acquisition timeout the system behaves as if a manual change to CHECKOUT mode was commanded. This means that the acquisition termination is synchronized (1PPS) and that no additional HV or state changes occur. The acquisition timeout period may be specified anywhere from 1 second to 65535 seconds (~18 hours). This timeout mechanism may also be used to command an acquisition of a predefined duration. Operationally this may facilitate commanding for acquisitions of pre-determined duration as no further commanding action is needed.

If during an acquisition a safety event occurs, the instrument will return to the 'SAFE' state and the ongoing acquisition will be terminated as quickly as possible. In this specific case the acquisition termination is not synchronized to the spacecraft time. This transition to the SAFE state terminates the acquisition, it will also result in the deactivation of the HV and when enabled (P_DoorEnable), the closing of the Aperture door (when one of the actuator power busses was active).

When the acquisition is terminated, the acquisition hardware will be commanded to stop the acquisition, and the contents of the current used acquisition buffer will be transferred to the spacecraft. In histogram and pixel list mode this means that the complete buffer contents are transferred, although in pixel list mode only a part of the memory will be filled with pixel list events (as indicated by a pointer in the housekeeping data (Hk_PIXEL_LIST_CNT)). In any case, this will also cause the Last Acquisition Completion in the HK telemetry to be updated, indicating the time at which the last acquisition was completed.

4.5.4 Test mode acquisition

In addition to the normal acquisition modes, the instrument supports test acquisition modes. Test mode and pattern are selected by a parameter of the 'start acquisition' telecommand. In these modes no actual detector data is acquired, but the acquisition memory is filled with known (deterministic) test patterns. Except from filling the memory with these patterns the complete nominal acquisition sequence is performed, including set up of the environment, most important:

HV will be activated if the system is configured to do so,

Aperture door will be commanded if enabled and will move if one of the actuator power busses is active.

This means that the nominal acquisition timing is very similar to the actual acquisition operation. But instead of sending the acquired science data via the high-speed link the instrument will send the known test patterns. For the histogram acquisition the complete nominal exposure/acquisition timing is followed. For the pixel list acquisition a small adaptation had to be made; instead of waiting for the acquisition memory to be filled, the exposure duration parameter (P_HISTO_EXP_DUR) is now also used to determine when the pixel list science data transfer should be started (like for the histogram acquisitions).

The generation of the different test patterns takes some time depending on the selected test pattern (see Appendix E). This test pattern generation starts when the normal acquisition would start (after setting up the conditions in the nominal sequence). As long as the test pattern generation is not completed, the science transfer is simply skipped. When the pattern generation is completed before the acquisition command is terminated, the first science transfer will start on the following first scheduled exposure duration. This test pattern generation function may be used for the various test of the high-speed science data transfer. Different known test patterns can be generated that support both low level interface tests and higher level end-to-end test of the science data transfer mechanism. The test pattern generation function fills both acquisition memory pages with a known pattern. Test patterns include:

- Incrementing 16-bit number pattern starting at zero
- Decrementing 16-bit number pattern starting at zero
- Simple 16-bit constant bit-alternating pattern 0xa5c3
- Leave acquisition memory contents as is
- Histogram like data, based on some tables and constants, including pulse height distribution and STIM pulses if enabled
- Pixel list like data, based on some tables and constants, including simulated time hacks at a fixed rate.

Note that the first word from the acquisition memory is never read by the transmission hardware; instead of the first memory word the identification word is transmitted (see Appendix D).

4.5.4.1 Histogram Test Pattern

Fill the indicated detector memory with a simulated histogram test pattern based on a hard coded single line spectrum with a spectral range from location 98 to 916. The rows of the test pattern are generated by scaling the values from the single spectral line with a one byte-scaling factor, resulting in histogram pixel values in the range of 0 to 65025. The first two and the last two rows of the histogram have scaling factors of zero so these entries will be empty and simulate the dark count areas in the normal histogram. The test pattern generation also adds two simulated STIM entries to the array when the STIM function was active during the commanded test acquisition. The simulated STIM entries consist of normal distributed events around the nominal STIM locations with a mean of 0.2 and a deviation of 1.2. The function will also add a pseudo pulse height distribution (PHD). The PDH will be added to the start of the second and third histogram line and consists of a 8000*251 times a simple normal distribution with a mean of 23 and a deviation of 12 limited to a value of 64005.

4.5.4.2 Pixellist Test Pattern

The simulated pixel list pattern contains time hacks every 31 entries and the hack counter uses the current value of the time hack setting to control the increment of hacks. The generated pixellist consists of eight fixed event-generating locations:

(2,112), (24,112), (30,358), (14,420), (9,585), (5,716), (5,718) and (14,912)

The actual generated simulated events vary around these eight points though using a simple triangle shaped distribution in the spectral dimension. In the spatial dimension the simulated events occur on a single line. The generated events vary around the 8 locations in a fixed offset sequence:

0,1,-2,-1,0,1,2,-1,0,-2,1,-1,0,2,1,0,-1,-3,-2,1,2,-1,0,1,3,-1

The purpose of this test pattern is the verification of the end-to-end transfer and processing chain and this test pattern can be processed by GSE as a normal pixel list data set. The pixellist test pattern does not include any STIM related entries.

4.6 Instrument Safety and Monitoring

The safety-checking function is always active and checks a number of parameters for out of limit conditions and reports the results of these checks in the housekeeping data. When an unmasked safety event is detected safing actions will be performed and will cause an autonomous commanding of the instrument to the SAFE state and the associate safing actions. To prevent this autonomous safing, when mission critical observations are being performed (e.g. during the Pluto and Charon solar occultation observations), the safety checking can be disabled (masked) to prevent possible termination due to accidental detection of a safety condition.

4.6.1 Safety checking

The LAMP instrument performs a number of different internal safety checks to protect the instrument from possible damage. These checks and actions are completely handled within the instrument to guarantee a quick response in case problems are detected. This monitoring uses various internally measured parameters to determine proper operation. The rate of the monitoring depends on the particular parameter checked, as different parameters become available at different rates.

There are three classes of monitored parameters for the internal safety checking:

HVPS parameters	Mcp Voltage, Anode Voltage and Strip Current measured by the safety task every 100 ms
Detector count rate	determined by the safety task every second
Temperatures	measured and checked by the heater task every second

If a safety condition in any of these three classes is detected, the safety processing determines if any action to protect the instrument should be taken.

4.6.1.1 HVPS parameter monitoring

Whenever the instrument is powered on, the instrument safety checking function will monitor the performance of the HVPSs. The main task of this monitoring is to prevent any damage to the instrument due to a problem with HVPS or detector. The immediate action taken, when an out of limit condition is detected by the safety checking function, is to deactivate the HVPSs. During critical science operations this safing action may be disabled to prevent accidental deactivation of the HVPS (and thus termination of the acquisition operation).

The safety checking function measures the HVPS parameters of both of the HVPSs every 100 ms and checks the measured values against predefined limits (see Table 13). If any of the verified parameters are outside these limits for a specified (configurable) number of consecutive measurements, the safety mechanism is triggered. All the used limits are stored in the parameter file. This provides for standard defined limits after instrument power-up but still offers the possibility of in-flight adaptations, for instance to compensate for aging or changed operational plans. Note that as the HK telemetry packets are only generated once every second the actual measurements triggering the safety condition may not be included in the telemetry data, but the housekeeping data provides identification of the current safety conditions (Hk_LAST_SAFETY).

For all three HV parameter checks a separate 'occurrence counter' checks that an out-of-limits condition only triggers the safety processing when the condition exists for more than a specified number of consecutive measurement cycles. This prevents measurement noise from triggering the safety handling. For instance when a limit of 5 is defined, it will be triggered when for five consecutive measurements (= 5*100 ms = 0.5 sec), an out of limit condition, is detected. The limit values for the 3 separate occurrence counters are specified in the parameter table (P_HV_FAIL_MCP, P_HV_FAIL_STRIP and P_HV_FAIL_ANODE).

Table 13: High Voltage safety checking

Parameter	Check	Description
MCP Voltage (largest read-back)	• HvSet ≤ Maximum Limit • When commanded HV is above minimum: | McpV - Expected HV | ≤ delta	Commanded High voltage not too large and matches commanded voltage
Strip Current (sum of read-backs)	• StripI ≤ Maximum Limit	Strip current not too large
Anode Voltage (largest read-back)	• AnodeV ≤ Maximum Limit • When commanded HV is above minimum: AnodeV ≥ Minimum Limit	Anode voltage within expected limits
The parameter values for these limit checks are obtained from the parameter table. The following parameters are used: • P_HV_LOW_SAFETY (Hv Setpoint above which HVPSs output reach nominal levels) • P_HV_MAX_HVSET (Maximum allowed Hv setpoint) • P_HV_MCP_TOL (Maximum Mcp voltage tolerance) • P_HV_MAX_STRIPI • P_HV_MIN_ANODEV and P_HV_MAX_ANODEV • P_DAC_ADC_FACTOR (converts from Hv setpoint to expected Hv readback value) Expected HV (readback) = HvSet * P_DAC_ADC_FACTOR		

During nominal HVPS rampup operations in (vacuum) ground testing the HVPS characteristics shown in have been observed. The detector resistance is temperature dependant so the measured HVPS output current will also be dependant on the temperature.

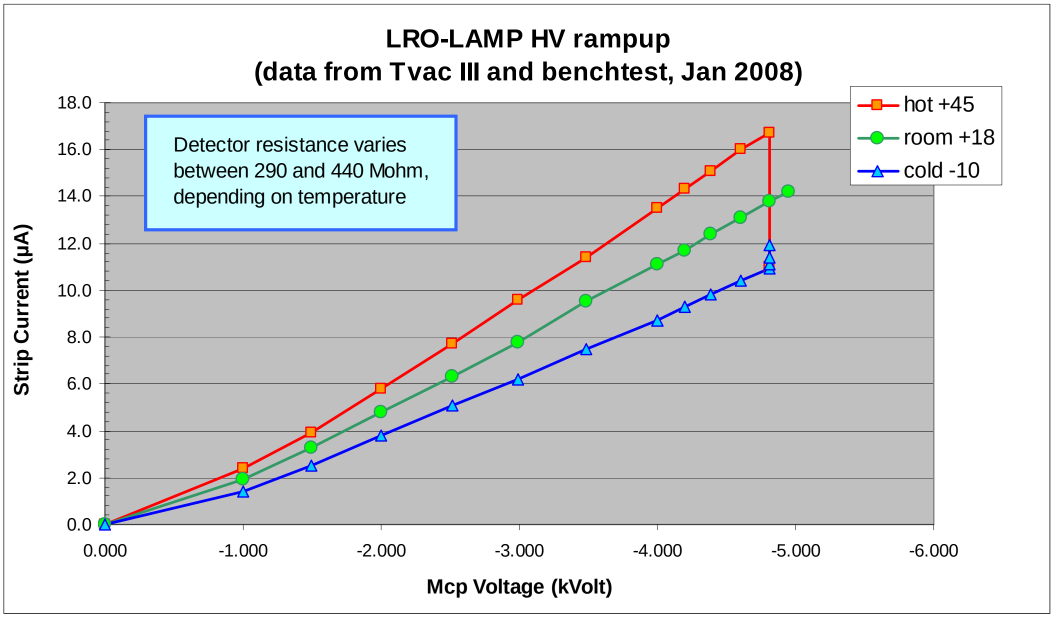

Figure 21: HVPS rampup characterization

4.6.1.2 Count rate monitoring

A bright object is determined by a measured high detector count rate exceeding a defined threshold. This is monitored as a safety condition as too much UV-light on the detector can damage the instrument (detector). The count rate value is calculated every second from the 24 bit raw-event counter value. The difference of two of the periodic readings from this counter is calculated and limited to a maximum of 65535 (16 bit value) and reported in the housekeeping data (Hk_COUNT_RATE). The count rate threshold is determined by a parameter (P_MAX_COUNT_RATE) and can be adjusted as needed to accommodate background counts (such as due to a "hot spot"). Note that the count rate measurement comes from the detector hardware (raw event counter) before the "hot pixel region" filters are applied.

4.6.1.3 Temperature monitoring

LAMP contains eight internal temperature sensors that are monitored at a one second rate. A sensor limit condition occurs if any monitored temperature sensor goes above a defined maximum safe limit. For each of the sensors, a separate maximum limit temperature is defined and stored in the parameter file (P_MAX_????_TEMP). If a sensor fails, the checking for that specific temperature channel can be disabled by masking the specific temperature check (P_TEMP_MASK), or by defining a maximum value for that specific limit check.

4.6.2 Safety processing

If any of the above mentioned safety conditions occurs, and the corresponding safety check is enabled, safety processing is triggered and the safety timeout timer is started (timeout period set to P_SAFETY_TIME). Also, if the safety override is not active, the instrument will immediately switch to the SAFE state. This transition to the SAFE state switches off the HVPSs, disables the heaters, terminates any actuator firing and, (when enabled (P_DoorEnable) when one of the actuator power busses is activated), closes the aperture door. During the safety timeout period the instrument will remain in the SAFE state (see Table 8). Thus, the software prevents any further operation or any actions that could damage the instrument (open door, high voltage on) until all of the safety conditions have disappeared for at least the timeout period.

Each of the safety checks can be individually masked by command, in which case that condition is ignored and is not used to determine the need for a shutdown. A general safety override is implemented which disables all safety actions and allows immediate operation of the instrument regardless of the safety condition(s) in effect. All safety condition information is included in the instrument housekeeping data (Hk_SAFETY_ACTIVE, Hk_LAST_SAFETY, Hk_SAFETY_TIMEOUT, Hk_SAFETY_ST, Hk_SAFETY_OVRD and Hk_SAFEMASK).

When a safety condition occurs during an acquisition, already acquired science data will not be lost. This case is handled as if a normal acquisition termination had occurred. Although the acquisition duration will be cut short (the actual acquisition termination time is reported in the housekeeping data (Hk_LAST_ACQ_DONE_TIME)), the already acquired data is not lost and will still be sent to the spacecraft.

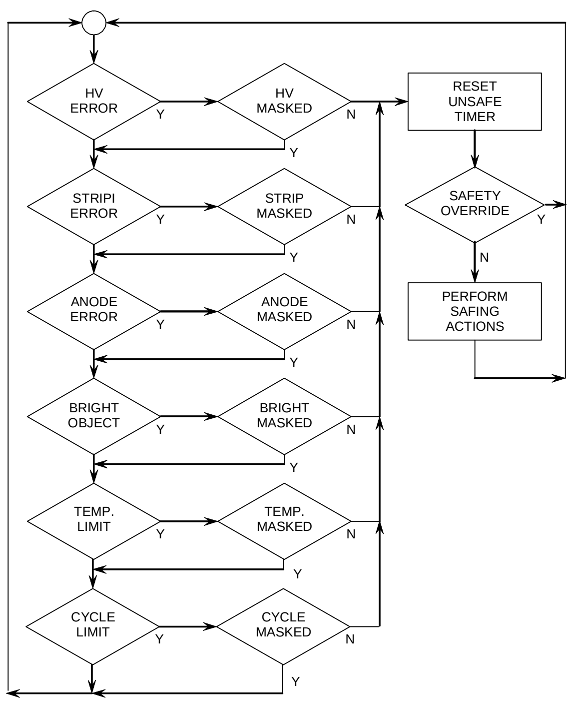

Figure 22: Safety Monitoring Algorithm Flowchart

Figure 22 shows a flowchart to demonstrate and clarify the operation of the safety algorithm. If any one of the safety conditions is true and it is enabled (not masked), then the "unsafe" timer will be reset. And, unless the safety override is in effect, actions will be taken to safe the instrument. The safing actions will command the instrument to the SAFE state and, therefore, include the following actions:

1. Deactivating the HVPSs,
2. Deactivate Actuator control,

3. Deactivating both decontamination heaters,
4. If enabled (P_DoorEnable) and if at least one of the actuator power busses is activated: close the aperture door.

4.6.3 WatchDog Timer

The onboard hardware includes a watchdog mechanism. The watchdog system is meant to catch catastrophic failures in the PERSI-LAMP Flight Software and, therefore, the software cannot disable this function. When the experiment on-board software does not trigger the hardware watchdog mechanism within a 4 second period, a hardware reset is issued and the on-board software restarts. Writing a specific byte pattern to the watchdog reset register performs this triggering. This mechanism prevents hanging software from permanently missing any control deadlines. The hardware reset ensures that the processor including all surrounding hardware circuitry is completely reset and returns to a (known) power-up state. This full reset should be enough to clear any specific software hang-up. The hardware reset will cause the HVPS to be turned off immediately, but no (hardware) command is issued to the aperture door control so the door may remain open.

The hardware provides a mechanism that counts the number of watchdog resets since the last power cycle. After multiple resets, this counter will eventually initiate the hardware only ‘state machine’ acquisition mode (see section 3.3.2). To support system test, the value of this watchdog expiration counter is reported in the housekeeping packet. Also a single special action is taken in order to use the spacecraft autonomy rules as a backup mechanism to prevent unintended ‘limp along’ mode activation. Whenever the number of watchdog resets exceeds 4 (fixed number, not a parameter, to keep the code and checking as simple as possible), the software will request a power cycle by the spacecraft by setting the ‘power-off’ request flag in the non-packetized telemetry. Explicit disabling of the corresponding spacecraft autonomy rule still allows for intentional overriding of this mechanism.

4.7 Memory functions

The instrument supports the three memory function telecommands that can interact with the various memory types in the C&DH

- Memory Check – calculates a checksum over a specified block of memory
- Memory Load – load a specified block of data into specified memory
- Memory Dump – dump a specified block of memory

Some memory functions are limited in that they don’t apply to all memory types.

The memory check function calculates a byte wise 16-bit xor rotate checksum over a specified block of memory. The result of the checksum calculation is reported in the housekeeping packet (Hk_MEM_CHECKSUM). At the start of the calculation, the result is set to zero and the reported value is not updated until the calculation is complete. At instrument startup automatically a checksum calculation is performed over the complete executing code and the result is reported in the housekeeping packet. The checksum function is used to verify the contents of an area of memory, for instance to verify the result of a load operation or as part of the periodic instrument checkout to check that the memory contents was not corrupted. The various checks performed by the instrument on the contents of the code memory (both at instrument startup and during execution of the code selftest function) use the same checksum algorithm. The result of the checksum calculation over the full code memory area (0-0x7f7d) is for each code page stored in the 16-bit word at code memory address 0x7f7e. The memory check function may be performed on all instrument memory types.

The memory load function copies the received data bytes directly into the specified memory. For the RAM based memory types this consists of a simple write to the indicated location with an appropriate setting of the page selection for the acquisition memory (only the inactive side of the acquisition memory can be written to). For the EEPROM memory, some additional preparations are needed to enable the write operation. This function uses the EEPROM block write mode, so only a single wait is needed to wait for the device internal EEPROM programming to complete (see 4.7.2). After completing any of the memory load functions, the just modified memory is read back and the two blocks are compared. Only when the two blocks are equal is a successful completion status returned, otherwise an error status will result. If the load function is used to load data to a memory area where no simple memory exists (e.g. special function registers) write and read may not result in the same data, which would result in an error message, although the action was actually successful. Memory load can be performed in all memory types except for the PROM memory that has a fixed content and can't be modified during flight.

The memory dump function reads the contents of the memory area and sends the contents of this memory to the spacecraft in the form of memory dump packets. A single memory dump command may generate multiple memory dump packets as each packet can only contain up to 128 bytes of data and a dump request could request much more data. Memory dump packets are only generated when the 'dump allowed' flag is set (included in the time synchronization message, see 4.3.1). When no memory dump packets are accepted, the memory dump function waits until the spacecraft again accepts packets. The memory dump function generates packets on the fly, but the software can easily generate packets with 128 bytes of data within one second. This means that a large memory dump operation can generate data sufficiently fast to add a memory dump packet to each generated ITF frame. The last generated packet of a memory dump sequence may contain less than the full 128 data bytes, but the packet is still a full size memory dump packet. There are four data structures defined in the LAMP flight software that are allocated at specific addresses and the memory dump command can be used to retrieve this data (Table 14).

Table 14: Special memory areas

Start Address	Size bytes	Storeage
0x8200	128	Error Log (8 entries)
0x8300	128	Parameter File (active copy, 128 byte size entries)
0x8400	128	Door Life result table (16 entries, cycles)
0x8500	256	Receive buffer (TC, 256 byte size entries)

4.7.1 EEPROM write

The EEPROM memory provides a non-volatile storage function that can be modified during flight. It is used in the instrument both to store the default values of parameter list and to enable (semi-permanent) updates of the instrument flight code. The hardware provides a paging mechanism that allows one of the four defined EEPROM pages to be moved into the code space ands replace the PROM memory that normally resides in that area. The four EEPROM pages all are a full 32 kbyte in size and can contain the complete flight software. At launch, the plan is to have all four EEPROM pages contain the same version flight software that is contained in the PROM memory as a backup.

The write function on the EEPROM memory always uses the memory in the block write mode. This limits the number of write cycles needed to change the contents of the memory. The write EEPROM function includes an appropriate wait period (10s of milliseconds) to wait until the internal device programming cycle is completed. During this period data read from any location of the EEPROM is not stable so this write function can only be called when code is being executed either from PROM (or RAM). The software checks for this and will not allow EEPROM writes when this constraint is not met.

4.7.2 EEPROM Limited life issues

The used EEPROM devices are only guaranteed for a limited number of write cycles. The devices are used in the block write mode which allows for writing up to 128 bytes in a single operation. When the devices are used in this block write mode the devices are guaranteed for up to 10000 write cycles per block including the appropriate derating factors. For write operations of the parameter list to the non-volatile parameter storage in EEPROM, the number of write cycles used is automatically maintained and stored within the EEPROM (specified location in the parameter file, see Appendix F). For code patches and updates, there is no automatic mechanism that counts the number of used EERPOM write cycles and a manual administration will be needed.

Another issue is the guaranteed data retention time of the EEPROM devices that is limited to 10 years (again after appropriate derating). The EEPROM memory in the system is used both for the storage of patched code (up to 4 complete pages) and for the parameter file (3 redundant copies). Failure of either of these functions is not directly fatal to the instrument operations. It would either reduce the provided redundancy (no ‘permanent’ code patches, although execution from RAM is still possible), or require some additional commanding (no parameter storage).

The allowed number of write cycles is not expected to be an issue for the EEPROM code storage. Even for the possible extended mission duration of 5 years, this number would allow for more than five updates every day. Still operational procedures will have to include a tracking of the number of commanded programming cycles (both pre- and post-launch). The data retention limitation can be handled by re-programming the devices at a certain point(s) in the mission. This reprogramming can be the result of a code patch operation (see section 4.7.4) or when the original PROM code is still in use; the PROM code may be used to ‘refresh’ the EEPROM contents. The system includes a special procedure to copy the PROM contents to all four EEPROM pages (see section 4.7.3).

The non-volatile version of the parameter file is expected to be updated more frequently, but still the number of allowed write cycles is expected to be much larger than the number of modifications expected during the mission. Temporary changes of parameter values only affect the RAM based (current working) version of the table and, thus, don’t count to the total number of write cycles. The parameter file will only be written to EEPROM when the system receives and explicit commanded to do so. To facilitate tracking of the number of write cycles a counter is maintained within the parameter file (see section 4.4.4) registering the total number of Parameter File write cycles. Nominal operations are expected to include a number of parameter file adjustments to include the results of the in-flight measurements in the acquisition parameters; also nominal operations are expected to include the setting of acquisition duration parameters and writing these into non-volatile storage. This expected nominal usage should satisfy the ‘refresh’ requirements to guarantee proper data retention. In addition, the triple redundant storage (see section 4.4.4) of the parameter file would provide an indication when problems start to develop.

4.7.3 Duplicate PROM

The instrument software includes a special function that can copy the contents of the current executing code memory to the four EEPROM pages. This procedure was used to prepare the system for the launch configuration in which all four EERPOM pages will contain an identical copy of the (PROM) flight software. During flight it may also be used to refresh the contents of all four EEPROM pages with either the contents of the PROM of one of the EEPROM pages. This function is not part of the nominal processing system started via the start program function at address 0x40dc (version 1.00 of the LAFS). Copying will take about 250 sec; during this period, the normal instrument processing is temporary suspended. This means that for this period no telemetry will be generated and no telecommands (including time messages) will be accepted. If telecommands (including time messages) are send during that period they will probably overflow the input hardware FIFO and will probably not be recognized.

4.7.4 Patch upload

Software patches may be uploaded using the load memory telecommand. Four EEPROM pages are available that can each store a complete version of the instrument software. A full software version may be uploaded in any of these pages.

The updated software can be delivered to the ground operation system in the form of an S-Record. Special conversion software has been created that will then converts this single S-record into a sequence of 255 memory load commands and corresponding confirm criticals. Each of these memory load commands will load 128 bytes into the EEPROM memory. The generated sequence will include small delays in between the commands to ensure that the system accepts all the telecommands and the check that ensures that not more than one telecommand per second is sent. As the spacecraft data handling software only implements a single buffer for this telemetry going to the PERSI-LAMP instrument, the commands all have to be spaced at least 2 seconds apart. This is not a limitation caused by the LAMP instrument; the processing of the memory load commands can easily be completed at a 1 Hz rate. Even with these 2 seconds delays though the complete upload of a full updated software version only will take 255 * 2 * 2 seconds (~17 minutes).

4.8 Internal Selftest

Instead of having a build-in test that is always executed on system startup, the instrument software includes a self-test that can be activated by a separate telecommand. The test executed at system startup is limited to a quick code memory check to verify that the core image is correct (see section 4.1.1). This check makes sense as a backup for the PROM code storage is available that may be activated to correct the problem. This results in a quick startup and provides for an extended test when required. In any case, failures detected by the self-test can’t automatically result in corrective actions. The single string nature of the instrument (C&DH) makes automatic recovery impracticable and in case of problems ground intervention is required (as some workaround may still exist). The internal self-test consists of a number of tests that can be performed within the environment of the executing real-time kernel. If more extensive test functions are required to diagnose a problem, the appropriate test functions have to be separately loaded either in RAM or EEPROM memory and can be executed through the ‘start program’ telecommand. Such extended test procedures need not be limited to running in the environment of the real-time kernel.

The execution of the different tests within the self-test function is controlled by the parameter of the telecommand which functions as a bit-field selecting the various tests. The results of the tests are reported in the housekeeping packet (Hk_TEST_STATUS). When the self-test command execution is started, all bits in this result byte will be set; when any of the steps completes without a failure, the corresponding bit in the result word will be cleared. All these tests are deterministic tests so beforehand the expected execution duration will be known for each of the steps.

Table 15: Self-test functions

Testcode	Test	Duration
0x01	Acquisition memory page 0	32 sec
0x02	Acquisition memory page 1	32 sec
0x04	Acquisition memory page window switching	7 sec
0x08	Free (unused) data RAM space	48 sec
0x10	Currently executing Code memory (checksum)	29 sec
0x20	Redundant parameters files (EEPROM)	<1 sec
0x40	EEPROM code memory (even pages, checksum)	48 sec
0x80	EEPROM code memory (odd pages, checksum)	48 sec

The acquisition memory test functions verify the memory within a given page by filling the complete page with a changing 16-bit pattern and then verifying the contents of all memory locations. The acquisition memory window switch test function verifies the paging mechanism by filling the first 1000 words on each page with a changing pattern that depends on the selected page and then verifying the contents of those 1000 locations on each page.

The RAM memory test can within the executing real time kernel only test the unused RAM memory. Given the small memory footprint of the application software, this means that still 86% of the available general purpose RAM can be verified, so this should provide a fairly good indication about the status of the RAM memory. The test fills the RAM memory with a 7 byte long pattern, first of a shifting 'one' and then of a shifting 'zero', in between and at the end the content of the memory is again verified. An odd pattern length was selected so the chances of missing a stuck data or address line are minimized. If a more extensive memory test or a test of the RAM used is required, a special purpose test function could be loaded in RAM before executing the test while the operational system is temporary suspended. In order to be able to restart the real-time kernel after completion of the test, the contents of the used RAM memory should be temporary saved in this case.

The code memory and the 4 EEPROM pages are checked by calculating the checksum over the code area (0-0x7f7d) and comparing these to the pre-calculated checksums that are stored in the code memory pages at address 0x7f7e. Normally this test will be successful as any update of the code in EEPROM should also include an update of the corresponding checksum (a full patch will include the update of the checksum). The parameter file test verifies the contents of the 3 redundant versions of the parameter file that is stored there and verifying that all three copies are identical.

Appendix A - Overview C&DH Electronics Hardware

As a reference an overview is provided of the C&DH Electronics hardware.

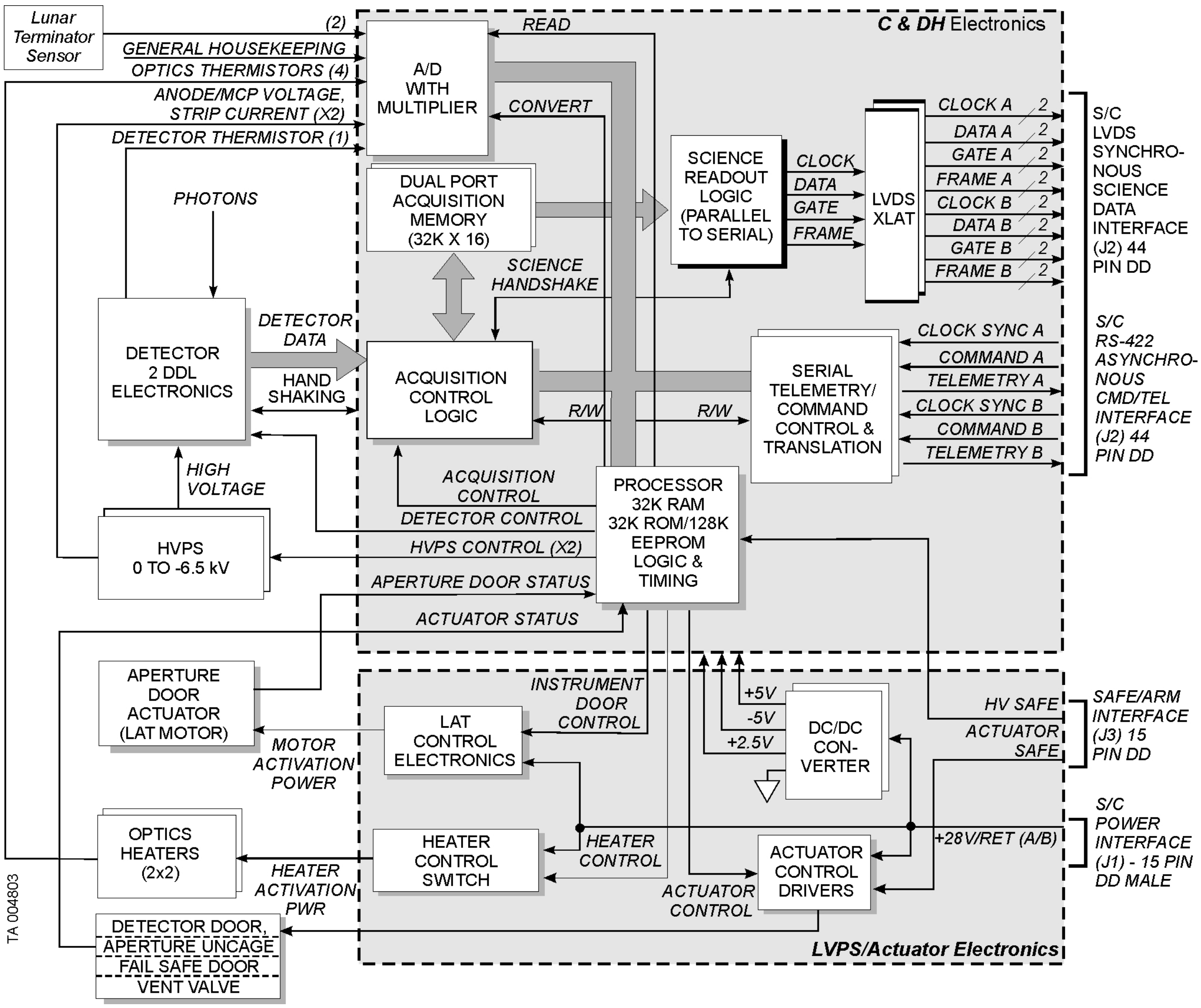

Figure 23: LAMP C&DH Electronics block diagram

Electronic block diagram notes:

- Acquisition memory consists of 2 pages each 32k x 16, one is accesses by the acquisition hardware, the other may after the Science readout be accessed by the processor,
- One of the general termistors is mounted on the SOC mirror,
- J2 consists of a single connector.

The power block diagram (Figure 24) shows the instrument configuration with the redundant instrument and actuator power busses. The redundant instrument power busses each power a separate Low Voltage

Power Supply (LVPS), which is directly coupled to a corresponding High Voltage Power Supply (HVPS). The instrument detector and electronics consist of a single string. On the actuator side, the redundant actuator/heater power busses power the corresponding actuators, except for the Aperture door motor, this is a single string element that may be powered by either actuator power bus.

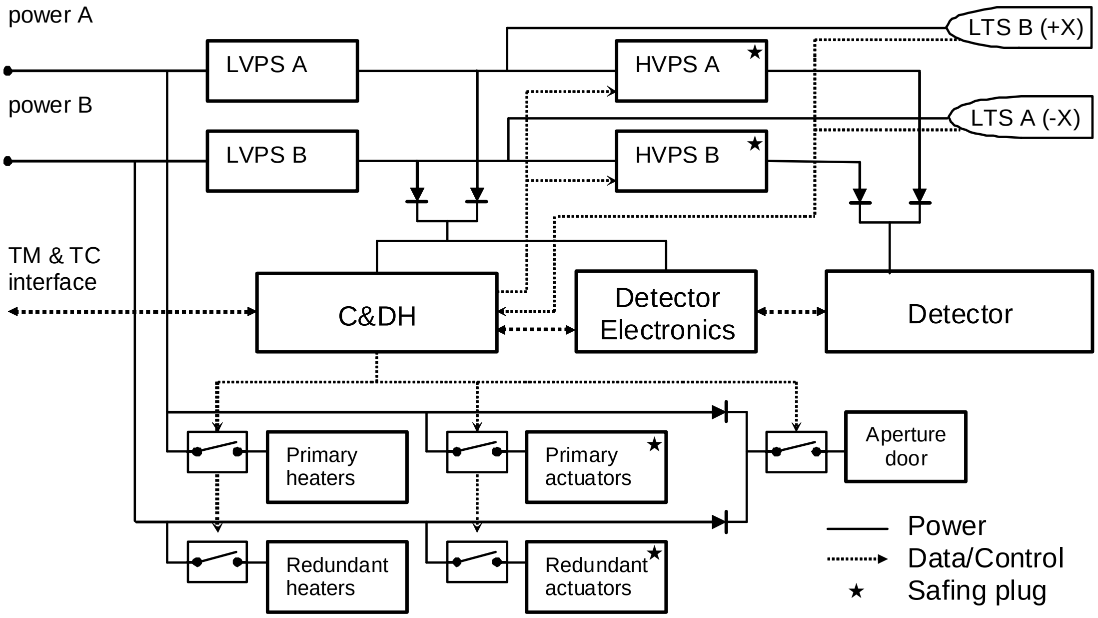

Figure 24: LAMP instrument power block diagram

Instrument power consumption

During integration testing power measurements on the FM model instrument were performed. The measurements are all based on a nominal bus voltage of 31 Volt, performed at room temperature.

Table 16: Instrument power consumption

	A-side only current	B-side only current	A+B side (total current)
Instrument idle	117mA	116mA	138mA/ 4.4W
HVPS on	+8mA	+7mA	+10mA / +0.3W
Mirror heater	+46mA/ +1.4W	+47mA/ +1.4W	+93mA/ +2.8 W
Grating heater	+46mA/ +1.4W	+47mA/ +1.4W	+93mA/ +2.8W
Aperture door activated (< 2 sec)	+70mA	+70 mA	+140mA/ +4.4W
WPA activated	+202mA	+198 mA	+379mA[1)] / +6.1W
SMA peak power (<100 ms)	+630mA/ +20W	+630mA/ +20W	+1260mA[1)]
Valve peak power (<50 ms)	+444mA/ +14W	+444mA/ +14W	N/A[2)]

1) possible but only during non-nominal operations.
2) valve opening and closing at the same time doesn't make sense.

Blue: measured on spacecraft @ 32Volt

The LAMP instrument can not measure it's own current draw or voltage of the S/C busses, but the S/C can. The LAMP instrument – spacecraft power wiring is such that the return current will share the returns of both power busses (due to a single return connection in the aperture door motor). Therefore the S/C current measurement circuitry will always report an A and a B side current even though only a single power bus is active. Each of the measurement circuits will report roughly half of the current.

Processor memory map

The processor of the C&DH is compatible with the 8051 processor except that the internal memory has been extended from 128 bytes to 256 bytes. This processor has a separate data and code spaces. The following memory map shows how the processor accesses these external memory spaces and the different peripherals (the processor internal memory is not included in this memory map).

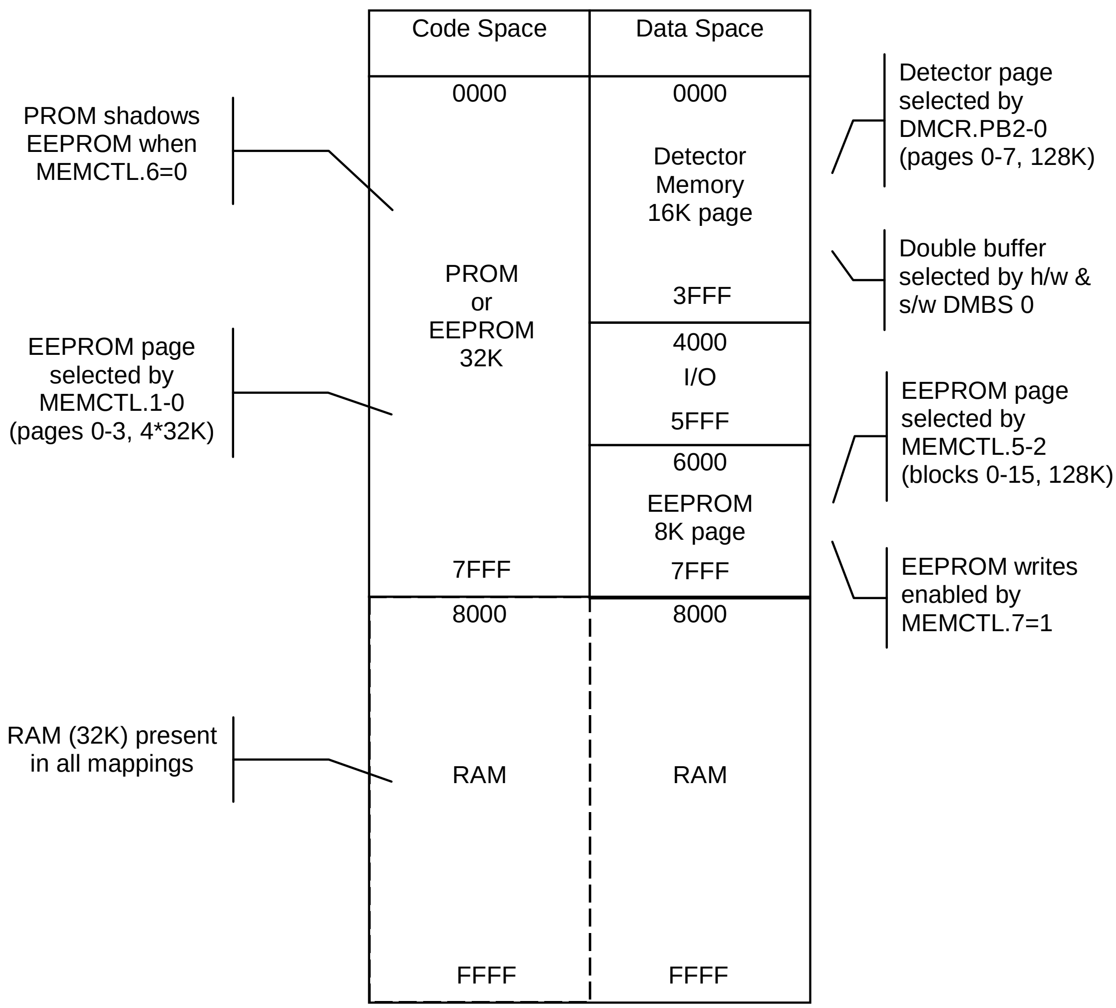

Figure 25: LAMP C&DH processor memory map

Appendix B - LAMP External Instrument Connectors

Table 17: LAMP Power Connector

Designation J1: 15-POSITION PIN				
Pin	**Description**	**AWG**	**Type**	**Mate**
1	CHASSIS			NC
2	30V_IN_A	22	Power Input	PSE
3	30V_RET_A	22	Power Return	PSE
4	CHASSIS			NC
5	SEC_RET(N/C)			NC
6	30V_RET_B	22	Power Return	PSE
7	30V_IN_B	22	Power Input	PSE
8	CHASSIS			NC
9	30V_IN_A	22	Power Input	PSE
10	30V_RET_A	22	Power Return	PSE
11	CHASSIS			NC
12	SEC_RET(N/C)			NC
13	SEC_RET(N/C)			NC
14	30V_RET_B	22	Power Return	PSE
15	30V_IN_B	22	Power Input	PSE

Table 18: LAMP Command and Data Connector

Designation J2: 44-POSITION HIGH-DENSITY SOCKET				
Pin	**Description**	**AWG**	**Type**	**Mate**
1	SYNC_CLK_A+[1PPS]	24	RS-422	C&DH
2	TEL_A+[Telemetry]	24	RS-422	C&DH
3	CMD_A+[Command]	24	RS-422	C&DH
4	SYNC_CLK_B+	24	RS-422	C&DH
5	TEL_B+	24	RS-422	C&DH
6	CMD_B+	24	RS-422	C&DH
7	Not Used			NC
8	SCI_FRM_B+	24	LVDS	C&DH
9	SCI_GATE_B+	24	LVDS	C&DH
10	SCI_CLK_B+	24	LVDS	C&DH
11	SCI_DAT_B+	24	LVDS	C&DH
12	SCI_FRM_A+	24	LVDS	C&DH
13	SCI_GATE_A+	24	LVDS	C&DH
14	SCI_CLK_A+	24	LVDS	C&DH
15	SCI_DAT_A+	24	LVDS	C&DH
16	SYNC_CLK_A-	24	LVDS	C&DH
17	TEL_A-	24	RS-422	C&DH
18	CMD_A-	24	RS-422	C&DH
19	SYNC_CLK_B-	24	RS-422	C&DH
20	TEL_B-	24	RS-422	C&DH
21	CMD_B-	24	RS-422	C&DH

Designation J2: 44-POSITION HIGH-DENSITY SOCKET				
Pin	**Description**	**AWG**	**Type**	**Mate**
22	Not Used			NC
23	SCI_FRM_B-	24	LVDS	C&DH
24	SCI_GATE_B-	24	LVDS	C&DH
25	SCI_CLK_B-	24	LVDS	C&DH
26	SCI_DAT_B-	24	LVDS	C&DH
27	SCI_FRM_A-	24	LVDS	C&DH
28	SCI_GATE_A-	24	LVDS	C&DH
29	SCI_CLK_A-	24	LVDS	C&DH
30	SCI_DAT_A-	24	LVDS	C&DH
31	[low-speed A]RET			
32	[low-speed B]RET			
33	CHASSIS			
34	CHASSIS			
35	Not Used			NC
36	Not Used			NC
37	[SCI_FRM_B]RET			
38	[SCI_GATE_B]RET			
39	[SCI_CLK_B]RET			
40	[SCI_DAT_B]RET			
41	[SCI_FRM_A]RET			
42	[SCI_GATE_A]RET			
43	[SCI_CLK_A]RET			
44	SCI_DAT_ARET			

Table 19: LAMP Safing Connector

Designation J3: 15-POSITION HIGH-DENSITY SOCKET identical to the Spacecraft Arm/Test Connector Panel				
Pin	**Description**	**AWG**	**Type**	**Mate**
1	HV1_SAFE	TBD	Safing Plug Loop-back	S/C Test Panel
2	[HV1_SAFE]RET	TBD	Safing Plug Loop-back	S/C Test Panel
3	SPARE			NC
4	SPARE			NC
5	SPARE			NC
6	ACTUATOR_SAFE1	TBD	Safing Plug Loop-back	S/C Test Panel
7	[ACTUATOR_SAFE1] RET	TBD	Safing Plug Loop-back	S/C Test Panel
8	ACTUATOR_SAFE2	TBD	Safing Plug Loop-back	S/C Test Panel
9	SPARE			NC
10	SPARE			NC
11	HV2_SAFE	TBD	Safing Plug Loop-back	S/C Test Panel
12	[HV2_SAFE] RET	TBD	Safing Plug Loop-back	S/C Test Panel
13	[ACTUATOR_SAFE2] RET	TBD	Safing Plug Loop-back	S/C Test Panel
14	SHIELD			
15	SHIELD			

Table 20: LAMP Safing Plugs

Ident	Safing Plug	Connections
-01	All Safe	1,2 6,7 8,13 and 11,12
-03	Actuator Safe	6,7 and 8,13
-05	HV Safe	1,2 and 11,12

Appendix C - Low Speed TM and TC Transfer Format

All low speed communication with the instrument uses ITF formatted (see Table 21) messages.

Table 21: Instrument Transfer Frame format

Byte #	Description	Value
0, 1 and 2	Frame Synchronization	0xfe, 0xfa, 0x30
3	Type of contained data:	0x01 - S/C time command 0x02 - telecommand 0x04 - telemetry
4	Checksum	Byte-wise XOR of all bytes following checksum field
5 – 6	Message Length	M - length of Message Data field in bytes
7 - 7+M	Message Data	Data (M bytes)

Table 22: Spacecraft time message data

Name	Size in bytes	Description
S/C Time	4	Time in seconds valid at the next 1-second (time sync) pulse
S/C Time fractional part	2	Fractional Time valid at the next 1 second pulse
Memory dump Allowed	1	0x00 – Memory dump allowed 0x01 – Memory dump not allowed

Table 23: Telecommand Message format

Name	Size in bytes	Description
Op-code	2	Instrument defined field to specify the TC function, for the general memory load and dump functions s/c defined values are used
Macro field (1 bit)	2	Unused for LAMP, set to zero
Command Word Count (15 bits)		Number of (32-bit) words in the command message including op-code, macro field, word count and checksum (= W)
Command Words	(W-2)*4	instrument defined command (parameter) words (32 bits)
Command Message (CM) checksum	4	32 bit exclusive-OR of all the words of the command message, not used / checked by LAMP

The TC receiving task will perform a number of low level tests based on the ITF format, LAMP will indicate any problems using the following error messages as reported in the Housekeeping Telemetry (LAST_FAIL_CODE). Note that these checks apply both to the Telecommand and Time Message ITF frames.

ITF format check error codes		
Errors:	0x01	ecINCORRECT_ITF_CHECKSUM_1
	0x02	ecINCORRECT_ITF_CHECKSUM_2
	0x03	ecUNSUPPORTED_ITF_MESSAGE_TYPE_1
	0x04	ecUNSUPPORTED_ITF_MESSAGE_TYPE_2
	0x05	ecTOO_LARGE_ITF_CMD_PACKAGE_1
	0x06	ecTOO_LARGE_ITF_CMD_PACKAGE_2
	0x07	ecTOO_SHORT_ITF_CMD_PACKAGE_1
	0x08	ecTOO_SHORT_ITF_CMD_PACKAGE_2
	0x09	ecINCORRECT_1ST_ITF_SYNC_BYTE_1
	0x0a	ecINCORRECT_1ST_ITF_SYNC_BYTE_2
	0x0b	ecINCORRECT_2ND_ITF_SYNC_BYTE_1
	0x0c	ecINCORRECT_2ND_ITF_SYNC_BYTE_2
	0x0d	ecINCORRECT_3RD_ITF_SYNC_BYTE_1
	0x0e	ecINCORRECT_3RD_ITF_SYNC_BYTE_2
	0x10	ecITF_ERROR_WAITING_1_2
	0x11	ecTC_BUFFER_OVERFLOW_1
	0x12	ecTC_BUFFER_OVERFLOW_2
	0x13	ecRX_FLUSH_FAILED_1
	0x14	ecRX_FLUSH_FAILED_2
	0x2c	ecTIME_MESSAGE_TOO_SHORT
	0x2d	ecTIME_MESSAGE_TOO_LONG

The TC receiving task only checks the match between the Telecommand ITF packet size and the number of command words as indicated in the Telecommand message format. Note that the Telecommand message format 32-bit checksum is not checked. Further checks apply to the specific telecommands and are listed in Appendix E.

Telecommand message format check errors		
Error:	0x22	ecWRONG_NUMBER_OF_COMMAND_WORDS

Appendix D - Low Speed Telemetry

The overall LAMP low-speed telemetry structure is shown in Table 24. This telemetry data is nominally generated every second (after a 2-3 second power on delay) whenever the instrument is active. This data is send to the spacecraft using the ITF header described in Table 21.

Table 24: LRO-LAMP Telemetry data

Name	Size in bytes	Description
Alignment filler	3	3 bytes filler to ensure that the Housekeeping packet starts at a convenient address, all 3 bytes will be set to zero.
Housekeeping packet	122	Embedded CCSDS formatted LAMP Housekeeping packet.
When requested: **Memory Dump packet**	0 or 148	Embedded CCSDS formatted Memory dump packet only generated / included when explicitly requested by telecommand and allowed by S/C C&DH, one memory dump request may result in multiple packets being generated. The packet size is fixed to 128 data bytes.
Total telemetry data size	125 or 273	Note this excludes the 7 byte ITF header

Numbers in brackets '()' indicate the size in bits of fields smaller than one byte. The next table specifies the LAMP housekeeping telemetry packet. The TM fields in the LAMP housekeeping packet (APID 081) excluding the fields in the header all start with 'ALHK'. The common header fields are prefixed with 'H081'. Where applicable '#' may be used to indicate a number or letter that selects between a number of similar items, the description will include the applicable range.

Table 25: LRO-LAMP CCSDS Housekeeping packet

Parameter (size in bits where applicable)	Size in bytes	Description
Primary Header (CCSDS)		
H081PKTVNO (3)	2	Fixed value = 0; designates a source packet
H081PCKT (1)		Fixed value = 0; designates a telemetry packet
H081SHDF (1)		Fixed value = 1; designates presence of secondary header
H081APID (11)		Fixed value = 129 (= 0x81); designates LAMP housekeeping packet [D-ICD]
H081SEGF (2)	2	Fixed value = 3; single packet dump: each LRO-LAMP housekeeping packet is completely self contained
H081CNT (14)		Continuous sequence count of all generated CCSDS housekeeping packets since startup (modulo 16384), the heart beat signal will be implemented as the least significant bit of this counter.
H081PLEN	2	Number of secondary header + data bytes (116) - 1 = 115 (fixed size Hk packet) [D-ICD]

Parameter (size in bits where applicable)	Size in bytes	Description
Secondary Header (CCSDS)		
H081TIME	6	Instrument time will nominally be synchronized to the S/C time as send to LAMP using the time synchronization message [D-ICD].
Data (CCSDS)		
General Instrument State		
Spare (1)		-
LAHKOPSTAT (3)		0, 4 - illegal (off) 1, 5 - checkout, 2, 6 - safe 3 - acquire-pixel, 7 – acquire-histo
LAHKSAFETYACT (1)	1	1 - safety (timeout) active
LAHKLSTSAFETYACT (3)		0 - No safety, 1 - Countrate, 2 - McpV, 3 - StripI, 4 - AnodeV, 5 – HV Cycle, 6 - Temperature; cleared on transition out of SAFE state
LAHKPWR# (2)		Power status for each LVPS (1 or 2): 1- active.
LAHKHVPS# (2)		Reported status of each HVPS: 1 - on
Spare (1)		-
LAHKTURNOFFREQ (1)	1	1 - request instrument shutdown by s/c
LAHKWPADRV (1)		1 - WPA activated (control signal)
LAHKWPASWTCH (1)		1 - WPA stroke switch (monitor signal) activated
LAHKHVPS#SAFE (2)		HVPS safing status for each HVPS (1 or 2): 1 - safing plug installed.
LAHKACTR#SAFE (2)		Actuator safing plug status for each bus (1 or 2) 1 - safing plug installed.
LAHKMIRRHTR# (2)	1	Commanded mirror heater (1 or 2) status: 1- heater on
LAHKGRATINGHT# (2)		Commanded grating heater (1 or 2) status: 1- heater on
TC/TM Interface Status		
LAHKCMDRCVD (1)		1 - Command received during last second
LAHKSYNCMSGRCVD (1)		1 - Valid Timesync message received during last second
LAHKSYNCPLSRCVD (1)		1 - Timesync pulse received during last second (A or B side)
LAHKCRITCMDPEND (1)	1	0 - none, 1 - TC pending
LAHKMEMDMPALLOWED (1)		1 - memory dump allowed, reflects status received in most recent S/C time message
LAHKTCIFSTAT (3)		Primary/redundant TC channel status: 1 – first byte both, 2 – next byte primary, 3 –next byte redundant, 4– first byte primary, 5 - first byte redundant.
LAHKCMDACPTCNT	2	Modulo 2^{16} count of commands accepted [1)]
LAHKCMDREJCNT	2	Modulo 2^{16} count of commands rejected [1)]
LAHKCMDEXECNT	2	Modulo 2^{16} count of commands executed [1)]
LAHKLASTCMDACPT	1	Id (low byte) of last accepted command
LAHKLASTCMDFAIL	1	Id (low byte) of last failed command
LAHKLASTFAILCODE	1	Last failure code command/execution
LAHKCRITCMDTIMEOUT	1	Remaining timeout period for a critical command
LAHKSPHPKTCONTENT (1)	2	These two bytes match the contents of the science packet telemetry header word (see appendix C); packet contents: 0 – pixellist, 1 - histogram
LAHKSPHMEM (1)		Acquisition memory: 0 – side-A, 1 – side-B

Parameter (size in bits where applicable)	Size in bytes	Description
LAHKSPHLASTBLK (1)		Last Block: 0 – intermediate frame, 1 – last frame
LAHKSPHHWACQ (1)		HW controlled acquisition (State Machine Mode)
LAHKSPHBLKNUM (12)		Least significant bits of the frame counter
Acquisition and Detector Status		
LAHKDETDORPOS (2)	1	Detector door position: 0 - illegal, 1 - open, 2 – not open, 3 - illegal
LAHKAPDORPOS (2)		Aperture door position: 0 - error, 1 - closed, 2 - open, 3 - between
Spare (1)		-
LAHKLTSDARK (1)		0 - light, 1 - dark
LAHKHVPS#1CMDSTAT (2)		Commanded state of each HVPS (1 or 2): 1 - on
Spare (1)	1	-
LAHKHACKRATE (3)		Current selected time hack rate: 0= 4 ms, 1= 8 ms, 2 = 16 ms, …, 7= 512 ms
Spare (1)		-
LAHKHSTMOVRFLW (1)		Overflow occurred in high speed science transfer
LAHKACQMEM (1)		Current acquisition memory: 0 – side A, 1 – side B
LAHKPXLSTIMSTAT (1)		0 - STIM off, 1 - STIM on
LAHKCNTRATE	2	Count rate per second (Hz), as measured during the last second (based on the analog event counter)
LAHKHVPSSETPT	1	DAC counts (0 kV – 6 kV: 4 ~ 200 V, 175 ~ 4.95 kV)
LAHKEVTCNT	3	Current value of the hardware detector analog event counter
LAHKTIMEHACKCNT	2	Most recent value of the time hack counter (at the time of the last 1PPS pulse)
LAHKPXLLSTCNT	2	Most recent value of the h/w pixel list pointer (indicates progress of pixel list acquisition)
LAHKEXPTIMEOUT	2	Remaining current histogram exposure time in seconds (duration of histogram based on value specified in parameter file)
LAHKLASTACQDONETIME	4	Time of the last acquisition completion (ping pong buffer swap or end of acquisition)
LAHKACQTIMEOUT	2	Remaining time in seconds of the acquisition timeout counter
LAHKMCP1V	1	Mcp Voltage 1; ADC counts (almost equal to Hv DAC counts) [2)], 152 ~ 4.95 kVolt
LAHKANODE1V	1	Anode Voltage 1; ADC counts (190 ~ 600 Volt) [2)]
LAHKSTRIP1I	1	Strip Current 1; ADC counts (105 ~ 8 μA, 190 ~ 12 μA) [2)]
LAHKMCP2V	1	Mcp Voltage 2; ADC counts (almost equal to Hv DAC counts) [2)]
LAHKANODE2V	1	Anode Voltage 2; ADC counts (190 ~ 600 Volt) [2)]
LAHKSTRIP2I	1	Strip Current 2; ADC counts (105 ~ 8 μA, 190 ~ 12 μA) [2)]
LAHKMAXMCPV	1	Maximum Mcp Voltage in the last second; ADC counts
LAHKMAXSTRIPI	1	Maximum summed Strip Current in the last second; ADC counts (107 ~ 10 8 μA, 150 ~ 14 μA) [2)]
LAHKDISCV	1	Discriminator level readback; ADC counts (V = n*3 / 255, 89 ~ 1.05 Volt)
LTS data and Status		
LAHKLTSALO (1)	1	Most recent LTS sensor A low threshold status

Parameter (size in bits where applicable)	Size in bytes	Description
LAHKLTSAHI (1)		Most recent LTS sensor A high threshold status
LAHKLTSBLO (1)		Most recent LTS sensor B low threshold status
LAHKLTSBHI (1)		Most recent LTS sensor B high threshold status
LAHKLTSREQUEST(2)		Most recent LTS current request status: 0 - no request, 1 - light, 2,3 - dark
LAHKLTSDELAYED (2)		Most recent LTS delayed request status: 0 - no request, 1 - light, 2,3 - dark
LAHKLTSARAW	2	LTS A sensor raw; ADC counts, most recent acquired full resolution (14 bits)
LAHKLTSBRAW	2	LTS B sensor raw; ADC counts, most recent acquired full resolution (14 bits)
LAHKLTSADATA#	10	Scaled LTS A sensor data, up to 10 samples (0-9) as acquired during the last second
LAHKLTSBDATA#	10	Scaled LTS B sensor data, up to 10 samples (0-9) as acquired during the last second
LAHKLTSSAFECYCLES	1	Number of LTS based HVPS safe cycles in the current acquisition (saturates at 255).
Temperatures and heater set points		
LAHKMIRRSETPNTTMP	1	Mirror heater set point; ADC counts (215 cnts ~ 45 ºC)
LAHKGRATINGSETPNT	1	
LAHKMIRRATMP	1	Mirror temperature primary sensor; ADC counts (168 cnts ~ 20 ºC, 181 cnts ~ 25 ºC)
LAHKMIRRBTMP	1	
LAHKGRATATMP	1	
LAHKGRATBTMP	1	
LAHKCDHELECTMP	1	
LAHKDETHOUSETMP	1	
Safety status reporting		
Spare (2)		-
LAHKTMPSAFETY (1)		Temperature safety; 1 - safety in effect
LAHKCYCLESAFETY (1)		HV Cycle safety; 1 - safety in effect
LAHKANODESAFETY (1)	1	Anode Voltage safety; 1 - safety in effect
LAHKSTRIPSAFETY (1)		Strip Current safety; 1 - safety in effect
LAHKHVSAFETY (1)		Mcp Voltage safety; 1 - safety in effect
LAHKBRIGHTSAFETY (1)		Bright Object (Countrate) safety; 1 - safety in effect
LAHKSAFETYTIMEOUT	2	Unsafe timer; remaining unsafe period in seconds, 0 means no safety active/pending
LAHKSAFETYOVRD (1)		1 - all safety handling is overridden (deactivated)
Spare (1)		-
LAHKTMPSAFEMASK (1)		Temperature safety mask; 1 - masked
LAHKCYCLESAFEMASK (1)		HV Cycle safety mask; 1 - masked
LAHKANODESAFEMASK (1)	1	Anode Voltage safety mask; 1 - masked
LAHKSTRIPSAFEMASK (1)		Strip Current safety mask; 1 - masked
LAHKHVSAFEMASK (1)		Mcp Voltage safety mask; 1 - masked
LAHKBRIGHTSAFEMASK (1)		Bright Object safety mask; 1 - masked

Parameter (size in bits where applicable)	Size in bytes	Description
Diagnostic data		
LAHKCODESTAT (4)	1	Current PROM/EEPROM executing code: 0-3 - Illegal, 4-7 - PROM, 8 - EEPROM page 1, 9 - EEPROM page 2, 10 - EEPROM page 3, 11 - EEPROM page 4, 12-15 - Illegal
LAHKHWVER (4)		Board (hardware) version id (0 - illegal, 1-L1, 2-L2,…,3-FM, 4-L3, 5-SIM,…,7-PROM,…), read from parameter file, used to distinguish source of received TM data
LAHKSWMAJORVER (4)	1	Software major version: Build number
LAHKSWMINORVER (4)		Software minor version: Version number
LAHKRXINT#OFFSTAT (2)	1	Interrupt disable for each TC receiver (1 or 2): 1 - disabled
LAHKSYNC#STAT (2)		1PPS received in last second for each TC receiver (1 or 2): 1 - received
LAHKFRMERR# (2)		Latched (set until cleared) UART frame error status for each TC receiver (1 or 2): 1 - frame error
LAHKOVRRUN# (2)		Latched (set until cleared) UART overrun status for each TC receiver (1 or 2): 1 - overrun
LAHKMEMCKSM	2	Checksum calculated in response to the last issued check memory command, initially this reports the power on (executing code) checksum.
LAHKPROCIDLE	2	Count of passes though the Rtx scheduler idle loop, measure of processor idle time.
LAHKPROCSCHED	2	Count of calls to Rtx scheduler (last second).
LAHKTESTSTAT	1	Self Test Status: Each '0' bit indicates that the corresponding test completed successfully or was not selected, a 1 indicates not completed or failed.
LAHKDEBUG	10	Variable debug information field, controlled by parameter file entry P_DebugTest (66) and reported in the HK field FirstDeleted: 0: TaskStack (FirstDeleted = 0 - 9 or 0xff): Value of stack pointer for all 10 active PAFS tasks: 0 - main task, 1 - safety monitor, 2 - TM gather task, 3 - HVPS control task, 4 - actuator fire task, 5 - aperture door task, 6 - TC interface task, 7 - TM interface task, 8 - heater task, 9 - slow task 1: TaskCounters (FirstDeleted = 0xf1): 0 – RTC (4 bytes) 4 – Smon (2 bytes) 6 – PPS count (2 bytes) 8 – Heat task cycles (1 bytes) 10 – Slow task cycles (1 bytes) 2: LTS-Times (FirstDeleted = 0xf2): LSB of the hacktime at LTS measurements
LAHKMINSTACK	1	Minimum amount of free stack space detected during the last second
LAHKFIRSTDEL	1	Task number of the first deleted task (0-9); set to 10 if no deleted task exists, indicates the use of the debug block: 0-9 or 0xff: TaskStacks 0xf1: TaskCounters 0xf2: LTS-Times

Parameter (size in bits where applicable)	Size in bytes	Description
LAHKSLOWTASKSTAT (3)	1	Slow task status: 0 - start, 1 - idle 2 - memory check 3 - memory dump 4 - memory load 5 - histrogram acquisition 6 - pixellist acquisition 7 - internal test
LAHKEXPMAXSTAT (1)		Watchdog expiration count above 15
LAHKEXPCNT (4)		Number of watchdog expirations occurred since the last (power-on) instrument reset, modulo 16
Parameter value reporting		
LAHKPARAMINDEX	1	Parameter Index: Last Requested parameter index
LAHKPARAMVAL	1	Parameter Value: Current value of last requested parameter
Packet conclusion		
LAHKPKTCKSM	2	Calculated packet checksum before sending HK data to S/C
Total HK packet size	122	Fixed size CCSDS packet

1) Counters start counting from zero at instrument startup and will wrap around when the maximum value is reached

2) These analog parameters are measured at a higher rate (multiple measurements per second) to support the instrument safety checking, the most recent measured value is reported.

The memory dump packets are meant for special fault diagnostics, and they will only be generated upon specific memory dump request in the form of the Memory Dump Telecommand. These packets are used to transfer the contents of the instrument on-board memory to the ground station. These transfers are included for contingency operations to verify the actual contents on the experiment on-board memory.

The generation of these packets is requested by the memory dump telecommand; it may be used to request the contents of all experiment on-board memory (PROM, acquisition memory, RAM and EEPROM pages 1 to 4). The size of the memory dump telemetry packets will always be 128 data bytes; if a smaller dump is requested still a full size dump packet will be generated. Multiple packets will be generated when a memory dump is requested that does not fit in a single dump packet. The format of the memory dump packet is described in Table 26; it follows the rules for a proper formatted CCSDS packet, and the standard described in the LAMP data ICD. If present, each ITF telemetry frame will contain a second complete CCSDS packet, but a complete memory dump may consist of a series of CCSDS packets. As each of the packets includes a complete description of the area of memory that is included in the packet, no additional sequence identification is needed.

Table 26: LRO-LAMP CCSDS memory dump packet

Parameter	Size in bytes	Description
Primary Header (CCSDS)		
H082PKTVNO (3)	2	Fixed value = 0; designates a source packet

Parameter	Size in bytes	Description
H082PCKT (1)		Fixed value = 0; designates a telemetry packet
H082SHDF (1)		Fixed value = 1; designates presence of secondary header
H082APID (11)		Fixed value = 130 (= 0x82); designates LAMP housekeeping packet [D-ICD]
H082SEGF (2)	2	Fixed value = 3; single packet dump: each LRO-LAMP housekeeping packet is completely self contained
H082CNT (14)		Continuous sequence count of all generated CCSDS housekeeping packets since startup (modulo 16384), the heart beat signal will be implemented as the least significant bit of this counter.
H082PLEN	2	Number of secondary header + data bytes (142) - 1 = 141 (fixed size Md packet) [D-ICD]
Secondary Header (CCSDS)		
H082TIME	6	Instrument time will nominally be synchronized to the S/C time as send to LAMP using the time synchronization message [D-ICD].
Data		
Start address	4	Start address of the dumped memory
Byte count	2	Number of valid bytes in the packet
Memory type	1	0x50 = RAM 0x51 = EEPROM page 1 0x52 = EEPROM page 2 0x53 = EEPROM page 3 0x54 = EEPROM page 4 0x55 = Acquisition Memory (currently inactive page) 0x56 = CODE_MEM (currently executing code)
Spare	1	Unused, set to 0
Data	128	Data bytes
Total Memory packet size	148	Fixed size 148 bytes

Appendix E - High Speed Science Telemetry

Science telemetry consists of 'raw' science data packets (frames) of 32768 16-bit words in size. Science telemetry packets are generated by the acquisition hardware and sent to the spacecraft via the dedicated LVDS interface [ICD 5.3].

To identify the Science packets, a single 16–bit word header starts the packets. The header is generated by the acquisition hardware and includes the information listed in Table 27. The complete header word of the most recent generated science TM packet shall be included in the housekeeping TM packet to allow for correlation between the two data streams (low and high speed) after receiving the data by the ground segment.

Table 27: P-LAMP Science packet telemetry header

Field	Size in bits	Description
Packet contents (msb)	1	0 - pixel list 1 - histogram
Memory	1	0 - ping (side A), 1 - pong (side B)
Last block	1	0 - intermediate block 1 - last block (acquisition cycle terminated)
HW acquisition	1	HW controlled acquisition (hardware limp along), overrides software control after 8 consecutive watchdog timeouts (see C&DH hardware documentation)
Block number	12	Least significant 12 bits of the frame number

The contents of the remainder of the data packets depends on the acquisition mode:

- Pixel List — Each word (32767) in the remainder of the frame either describes a photon event or a time hack. For photon events bit 15 is set to zero and time hacks are indicated by bit 15 with a value of one. A photon event encodes the location of the detected event consisting of a 10-bit encoded spectral location and a 5-bit encoded spatial location in the remaining 15 data bits. The time hack is used to provide temporal information about the photon events. The acquisition hardware will generated and insert time hacks in the frame on a periodic basis (configurable for each acquisition in a range of 4 – 512 ms). In the remaining 15 bits, the time hack contains an incrementing counter that counts the number of 4 ms periods. This value allows for data recovery in case of lost frames or packets.
- Histogram — Each word (32767) in the remainder of the frame contains the number of photon events detected for that specific spectral/spatial location. This corresponds to considering the full frame as a two dimensional array of 32 (spatial) * 1024 (spectral) 16-bit counters stored row-wise (corresponding to the spectral dimension). These counters saturate at a maximum value of 65535 to indicate a completely filled counting bin, meaning that the counters do not wrap around. In addition some special data words (header and pulse height distributions) are overlying the lower left-hand corner of the actual array in a block of 4 (spatial) * 32 (spectral) words. This usage doesn't limit science data contents as no detector events occur in this region. Of this over-written block, the first row contains the header word and the second and third row contain the 64 words of pulse height information. The remainder of this block is filled with zeros. The pulse height

information is again a list of counted occurrences, but for the pulse height (event amplitude) of the separate events. Pulse height is available in 6-bit resolution resulting in 64 bins. These counters also saturate to a value of 65535. So a single photon event is counted both in the spectral/spatial array and in the pulse height list.

For test purposes the instrument can fill the memory with known deterministic pattern so the interfaces to the spacecraft and ground can be verified. The instrument software allows for the generation of 5 different test patterns.

The spacecraft will store the LAMP High Speed Telemetry data LAMP acquisition files before sending the data to the ground. The on board created files will be stored in the 'SSR/LAMP/' directory on the SSR. For nominal operations one file will be created for each acquisition which corresponds to a single orbit and that file will contain many acquisition frames each preceded by a 6 byte time stamp that indicates when the frame was received from the LAMP instrument.

Table 28: LRO-LAMP Science file format

Parameter	Size in bytes	Value		Description
LRO File Header				one header per file
File Type ID	4	Sci	HK	
		200	201	CRaTER
		210	211	Diviner
		220	221	LAMP
		230	231	LEND
		240	241	LOLA
		250	259	LROC (TBR)
		260	261	Mini-RF
			TBR	Table dumps
			TBR	Memory dumps
			TBR	Orbiter Housekeeping
			TBR	Orbiter Events
Spare	4	0		Alignment or Expansion
Start Time (seconds)	4	0-7FFFFFFF		Spacecraft time when the file was opened for writing
Start Time (sub-seconds)	4	0-FFFFFFFF		Spacecraft time when the file was opened for writing
Stop Time (seconds)	4	0-7FFFFFFF		Spacecraft time when the file was closed*
Stop Time (sub-seconds)	4	0-FFFFFFFF		Spacecraft time when the file was closed*
FileName	40	ASCII (with 0 padding)		Filename of the file as it appeared onboard the S/C (including path)
Acquisition Frames				Possibly many frames per file
Frame Generation Time	6			Time at which the frame was received by the spacecraft from the LAMP instrument
LAMP science data frame	65536			LAMP generated HS frame
Frame Generation Time	6			Time at which the frame was received by the spacecraft from the LAMP instrument
LAMP science data frame	65536			LAMP generated HS frame
Frame Generation Time	6			Time at which the frame was received by the spacecraft from the LAMP instrument
LAMP science data frame	65536			LAMP generated HS frame
	etc.			

Appendix F - Telecommands

LAMP operations are controlled by a set of 25 telecommands as listed in Table 29. In the remainder of this appendix, the functionality of these telecommands are described in detail, including the criteria that allow correct execution and the reported status. The instrument only accepts a number of these commands whenever the instrument is not in the SAFE state. To prevent inadvertent activation of a number of critical functions a critical command mechanism is implemented. For these so called 'critical commands' a separate confirmation command is needed before the execution of the actual command start. Note that this mechanism is completely separate from the critical command handling as implemented by the New Horizons ground system.

Table 29: P-LAMP Telecommands

Op-code (hex)	Mnemonic	Telecommand	Critical
	General commands		
0x6601	LANOOP	No Operation	N
0x6602	LASAFEMODE	Enter Safe State (Mode?)	N
0x6603	LACHKOUTMODE	Enter Checkout State (Mode?)	N
0x6604	LACRITCONF	Confirm Critical Command	N
0x6605	LASTARTHISTGRM	Start Histogram Acquisition	Y
0x6606	LASTARTPIXELLST	Start Pixel List Acquisition	Y
0x6607	LAPARAMSET	Set Parameter	Y
0x6608	LAPARAMSWRIT	Store Parameters	Y
0x6609	LAPARAMSLD	Load Parameters	N
0x660a	LAACTPIXSTIM	Set Pixel Stimulator On	N
0x660b	LADEACTPIXSTIM	Set Pixel Stimulator Off	N
0x660c	LADISCSET	Set Discriminator Level	N
0x660d	LAAPDORCLS	Close Aperture Door	N
0x660e	LAHVPSDEACT	Set High Voltage Off	N
	Not in SAFE state		
0x660f	LAAPDOROPN	Open Aperture Door	Y
0x6610	LAHVPSACT	Set High Voltage On	Y
0x6611	LAHTRCTRL	Control Heater	N
0x6612	LAWPASMAACT	Activate WPA/SMA	Y
0x6613	LASELFTEST	Perform Self Test	N
0x6614	LAPRGSTRT	Start Program	Y
0x6615	LAREQTURNOFF	Request instrument power down	Y
0x6616	LARSTTCSTAT	Reset TC status	N
	Memory functions (only inCHECKOUT state)		
0x6617	LAMEMCHK	Check Memory	N
0x6618	LAMEMLD	Load Memory	Y
0x6619	LAMEMDMP	Dump Memory	N

After receiving a complete telecommand and passing the ITF frame checks (see Appendix B), a number of general checks are performed to verify consistency of the received command bytes and to verify that the instrument is in a proper state to receive the command. As a result of these checks a command may be rejected as indicated by incrementing the rejected command counter, which is reported in the Housekeeping data. In addition to this reporting an error code and if available a code indicating the offending command will also be reported in the housekeeping data (LAST_CMD_FAILED and LAST_FAIL_CODE). These general error codes are listed in the following table and are not repeated in the error codes related to the specific telecommands (the specific description will list 'none' if no command specific error codes are applicable).

General telecommand check error codes		
Errors:	0x20	ecIMPROPER_LENGTH_FOR_COMMAND
	0x21	ecUNKNOWN_UNRECOGNIZED_TELECOMMAND
	0x22	ecWRONG_NUMBER_OF_COMMAND_WORDS
	0x23	ecCOMMAND_NOT_IN_SAFE_STATE
	0x24	ecALREADY_CRITICAL_COMMAND_PENDING
	0x25	ecINCORRECT_CONFIRM_CRITICAL
	0x26	ecMISSING_CONFIRM_CRITICAL
	0x27	ecNO_CRITCAL_COMMAND_PENDING
	0x28	ecCRITICAL_COMMAND_TIMEOUT

The remainder of this appendix provides a detailed description of all the PERSI-LAMP telecommands; this includes a description of the command parameters and their effects and the criteria governing the acceptance or rejection and execution reporting related to these telecommands. The acceptance, rejection and execution of a telecommand are indicated by the corresponding 3 counters that are included in the PERSI-LAMP housekeeping telemetry. A detected error is also reported in the housekeeping data in the filed 'last error' and the command that caused the error is included in the last failed command.

Mnemonic:	LANOOP		
Command code:	0x6601	**Critical Command:**	No
Command size:	8 bytes	**Disabled in SAFE state:**	No
Description:	Command without any significant action used to verify TC path, command will always be accepted and executed successfully		
	ident	**size**	**description**
Parameters:	none	-	-
Acceptance:	always		
Execution:	always		
Errors:	none		

Mnemonic:	LASAFEMODE		
Command code:	0x6602	Critical Command:	No
Command size:	8 bytes	Disabled in SAFE state:	No
Description:	Change to safe state (Mode?). This includes deactivation of HV, deactivation of one time actuator control, deactivation of decontamination heaters and closing aperture door (if enabled). If an acquisition was active at the moment of this command this acquisition will be stopped at the next timesync pulse.		
	ident	**size**	**description**
Parameters:	none	-	-
Acceptance:	always		
Execution:	always		
Errors:	none		

Mnemonic:	LACHKOUTMODE		
Command code:	0x6603	Critical Command:	No
Command size:	8 bytes	Disabled in SAFE state:	No
Description:	Change to checkout state (Mode?)is only possible when currently no safety condition is pending (or the safety override is active). If an acquisition was active at the moment of this command this acquisition will be stopped at the next timesync pulse.		
	ident	**size**	**description**
Parameters:	none	-	-
Acceptance:	Telecommand only is accepted if the change to CHECKOUT state is completed or if the delayed change to CHECKOUT state is scheduled (when acquisition was in progress).		
Execution:	Change to the CHECKOUT state may fail if a safety condition is in effect (safety timeout > 0) and safety override is not active, command will not be accepted in that case either.		
Errors:	0x30 ecFAILED_TO_CHANGE_TO_CHECKOUT		

Mnemonic:	LACRITCONF		
Command code:	0x6604	Critical Command:	No
Command size:	12 bytes	Disabled in SAFE state:	No
Description:	Critical commands need to be confirmed by the next command. This command must be a confirm critical command and it must be received within the timeout period. The fact that a critical command is pending will be indicated by housekeeping parameter 'CRIT_CMD_PENDING_ST'. The timeout period is determined by configuration parameter 'P_CMD_TIMEOUT' and the remaining timeout is reported in housekeeping parameter 'CRIT_CMD_TIMEOUT'. When a pending critical command times out an error code is generated, but the failing command is not reported.		
	ident	**size**	**description**
Parameters:	CONFIRMED_COMMAND unused	2 bytes 2 bytes	Command code of command to be confirmed -
Acceptance:	A confirm critical command is accepted when a critical command is pending and when the parameter of the confirm command matches the command code of the pending critical command. After that the specific rules for acceptance of the confirmed (critical) command have to be satisfied (see separate (critical) command descriptions).		
Execution:	The confirm critical command itself is never executed but the command that is being confirmed will (re-)start execution when the confirm critical is received and this may lead to successful command completion. This execution is described in the separate (critical) command descriptions.		

Mnemonic:	LACRITCONF	
Errors:	0x25	ecINCORRECT_CONFIRM_CRITICAL
	0x26	ecMISSING_CONFIRM_CRITICAL
	0x27	ecNO_CRITCAL_COMMAND_PENDING
	0x28	ecCRITICAL_COMMAND_TIMEOUT

Mnemonic:	LASTARTHISTGRM		
Command code:	0x6605	**Critical Command:**	Yes
Command size:	12 bytes	**Disabled in SAFE state:**	No

Description: Start acquisition in histogram mode, acquisition parameters are taken from internal parameter file, acquisition continues until commanded off (by commanding safe or checkout state) or until the parameter file defined timeout is reached. Test mode selection results in a deterministic test pattern instead of acquired data in the science TM packets.
In preparation for the acquisition the following actions are completed:

- Set discriminator, STIM and hot-segment registers to parameter file defined values.
- Command aperture door to position requested in the command parameter
- Command HVPS to the level indicated in the parameter file
- Clear the acquisition memory (unless test mode is selected)

Start of the actual acquisition is synchronized to the next sync pulse. During acquisition the time remaining until acquisition timeout and the remaining exposure timeout are reported in the housekeeping data.
When operating in test mode (MODE_SELECT>0) the normal actions and timing of the acquisition are maintained, but the science TM generation is skipped until the test pattern generation is completed.

	ident	**size**	**description**
Parameters:	APDOOR_POSITION	1 byte	0 – Door Closed 1 – Door Open
	MODE_SELECT	1 byte	0 – actual acquisition 1 – Incrementing number test pattern (~30 sec) 2 – Decrementing number test patter (~30 sec) 3 – Constant value (0xa5c3) test pattern (~23 sec) 4 – Leave acquisition memory as is (0 sec) 5 – pseudo histogram test pattern (~46 sec) 6 – pseudo pixel list test pattern (~40 sec)
	-	2 bytes	unused

Acceptance: Always; command is stored when waiting for ‘Confirm Critical’. ‘Confirm Critical’ is accepted when the acquisition start is accepted for execution by the ‘slow’ task when:

- Parameter values are within limits (door & mode)
- Slow task is available for execution (no acquisition, test or memory function active).
- Instrument can change to acquire state (no safety pending).

Execution: After correct confirmation and after reaching specified HV level when the actual acquisition is started or when a test mode acquisition is commanded and the generation of the specified test pattern is complete.

Mnemonic:	LASTARTHISTGRM	
Errors:	0x51	ecNO_ACQ_START_SAFETY_PENDING
	0x52	ecNO_ACQ_START_SLOW_TASK_BUSY
	0x53	ecNO_ACQ_START_WRONG_TEST_PARAM
	0x54	ecNO_ACQ_START_WRONG_DOOR_PARAM
	0xa0	ecSYNCED_HISTO_ACQUISITION_FAILED
	0xa2	ecCLEAR_HISTO_ACQUISITION_FAILED
	0xa4	ecSTART_HISTO_ACQUISITION_FAILED
	0xa6	ecACQUISITION_TIMEOUT_OCCURRED

Mnemonic:	LASTARTPIXELLST		
Command code:	0x6606	**Critical Command:**	Yes
Command size:	12 bytes	**Disabled in SAFE state:**	No
Description:	Start acquisition in pixel list mode, acquisition parameters are taken from internal parameter file, acquisition continues until commanded off (by commanding safe or checkout state) or until the parameter file defined timeout is reached. Test mode selection results in a deterministic test pattern instead of acquired data in the science TM packets. In preparation for the acquisition the following actions are completed: – Set discriminator, STIM, hack timer and hot-segment registers to parameter file defined values. – Command aperture door to position requested in the command parameter – Command HVPS to the level indicated in the parameter file – Clear the acquisition memory (unless test mode is selected) Start of the actual acquisition is synchronized to the next sync pulse. During acquisition the time remaining until acquisition timeout is reported in the housekeeping data. When operating in test mode (MODE_SELECT>0) the normal actions and timing of the acquisition are maintained, except that exposure duration parameter is used to initiate the science TM generation, also the science TM generation is skipped until the test pattern generation is completed.		

	ident	**size**	**description**
Parameters:	APDOOR_POSITION	1 byte	0 – Door Closed 1 – Door Open
	MODE_SELECT	1 byte	0 – actual acquisition 1 – Incrementing number test pattern (~30 sec) 2 – Decrementing number test patter (~30 sec) 3 – Constant value (0xa5c3) test pattern (~23 sec) 4 – Leave acquisition memory as is (0 sec) 5 – pseudo histogram test pattern (~46 sec) 6 – pseudo pixel list test pattern (~40 sec)
	-	2 bytes	unused
Acceptance:	Always; command is stored when waiting for ‘Confirm Critical’. ‘Confirm Critical’ is accepted when the acquisition start is accepted for execution by the ‘slow’ task when: – Parameter values are within limits (door & mode) – Slow task is available for execution (no acquisition, test or memory function active). – Instrument can change to acquire state (no safety pending).		
Execution:	After correct confirmation and after reaching specified HV level when the actual acquisition is started or when a test mode acquisition is commanded and the generation of the specified test pattern is complete.		

Mnemonic:	LASTARTPIXELLST	
Errors:	0x51	ecNO_ACQ_START_SAFETY_PENDING
	0x52	ecNO_ACQ_START_SLOW_TASK_BUSY
	0x53	ecNO_ACQ_START_WRONG_TEST_PARAM
	0x54	ecNO_ACQ_START_WRONG_DOOR_PARAM
	0xa1	ecSYNCED_PIXEL_ACQUISITION_FAILED
	0xa3	ecCLEAR_PIXEL_ACQUISITION_FAILED
	0xa5	ecSTART_PIXEL_ACQUISITION_FAILED
	0xa6	ecACQUISITION_TIMEOUT_OCCURRED

Mnemonic:	LAPARAMSET		
Command code:	0x6607	**Critical Command:**	Yes
Command size:	12 bytes	**Disabled in SAFE state:**	No
Description:	Set specified internally stored parameter to specified byte value. As one of the parameters is the safety mask, this TC is also used to change the active safety mask. Also the parameter reported in the HK packet is specified itself as a parameter, so this is also changed using this command. The value of the parameter value cannot be checked, as the different parameter file entries have different valid ranges.		
	ident	**size**	**description**
Parameters:	PARAMETER_INDEX	1 byte	Range: 0 – 58, see Appendix F
	PARAMETER_VALUE	1 byte	Range 0x00 – 0xff, meaning dependant on the specific parameter value
	unused	2 bytes	-
Acceptance:	Direct; command is stored when waiting for ‘Confirm Critical’. ‘Confirm Critical’ will be accepted when the parameter id is within the valid range.		
Execution:	After correct confirmation, when parameter id is within range and the specified value has been stored successfully.		
Errors:	0xb0 ecPARAMETER_INDEX_OUT_OF_RANGE 0xb1 ecPARAMETER_READBACK_FAILED		

Mnemonic:	LAPARAMSWRIT		
Command code:	0x6608	**Critical Command:**	Yes
Command size:	8 bytes	**Disabled in SAFE state:**	No
Description:	Store parameter table into non-volatile EEPROM memory (all three copies), before performing the store operation the counter indicating the number of table write operations will be incremented. When multiple storage failures (in different EEPROM pages) are detected only the last detected failure will be reported. For a storage error only the page is indicated, memory dump commands could be used to investigate this problem further.		
	ident	**size**	**description**
Parameters:	none	-	-
Acceptance:	Direct; command is stored when waiting for 'Confirm Critical'. 'Confirm Critical' is accepted when the currently code from PROM is being executed.		
Execution:	After correct confirmation, when currently code from PROM is being executed, all the three store operations have been completed and the correct storage has been verified.		
Errors:	0xb2 ecSTORE_PARAMS_READBACK_FAILED_1 0xb3 ecSTORE_PARAMS_READBACK_FAILED_2 0xb4 ecSTORE_PARAMS_READBACK_FAILED_3		

Mnemonic:	LAPARAMLD		
Command code:	0x6609	**Critical Command:**	No
Command size:	12 bytes	**Disabled in SAFE state:**	No
Description:	Load parameter table from non-volatile memory into operational table, restores power-on condition, parameter allows for restore from all redundant stored tables using a majority vote, restore from a selected single table or initialize to a hardcoded default set. When the parameter file is loaded from a selected single table the instrument state will always change to SAFE. When the parameter file is loaded using majority voting from the 3 EEPROM stored tables the following actions are taken, for each of the byte parameters in the parameter file: – All 3 agree: Use the value for the parameter – Two agree: Use the value of the two tables that agree for the parameter value, report the problem – All 3 different: Command fails, don't change the parameter value, go to SAFE state, report the error		
	ident	**size**	**description**
Parameters:	TABLE_SOURCE	1 byte	0 – load from all three tables (majority vote) 1 – load only from table 1 2 – load only from table 2 3 – load only from table 3 17 – load from hardcoded (PROM) table, see values listed in Appendix F.
	unused	3 bytes	
Acceptance:	When a correct table source is specified		
Execution:	When a single table source is specified always, when a majority vote is requested only when no errors are detected.		
Errors:	0xb6 ecINCORRECT_TABLE_FOR_PARAM_LOAD 0xb7 ecLOAD_PARAM_TABLE_1_MISMATCH 0xb8 ecLOAD_PARAM_TABLE_2_MISMATCH 0xb9 ecLOAD_PARAM_TABLE_3_MISMATCH 0xba ecLOAD_PARAM_ALL_DIFFERENT		

Mnemonic:	LAACTPIXSTIM			
Command code:	0x660A		**Critical Command:**	No
Command size:	8 bytes		**Disabled in SAFE state:**	No
Description:	Command stimulator on, the current commanded state of the pixel STIM is reported in housekeeping parameter 'PIXEL_STIM_ST'. The pixel STIM may either be activated by direct telecommand or as a side effect of the commanded start of an acquisition at which memory the parameter file entry 'P_StimEnable' determines whether STIM is commanded on or off. Executing this command will also set the entry 'P_StimEnable' in the active copy of the Parameter File, so a commanded acquisition start after any manual commanding will use the last commanded STIM state, but after an instrument reset the EEPROM value will be used.			
	ident	**Size**	**description**	
Parameters:	none	-	-	
Acceptance:	always			
Execution:	when hardware control is successfully activated			
Errors:	0x89 ecSTIM_NOT_ACTIVATED			

Mnemonic:	LADEACTPIXSTIM			
Command code:	0x660B		**Critical Command:**	No
Command size:	8 bytes		**Disabled in SAFE state:**	No
Description:	Command stimulator off, the current commanded state of the pixel STIM is reported in housekeeping parameter 'PIXEL_STIM_ST'. The pixel STIM may either be activated by direct telecommand or as a side effect of the commanded start of an acquisition at which memory the parameter file entry 'P_StimEnable' determines whether STIM is commanded on or off. Executing this command will also clear the entry 'P_StimEnable' in the active copy of the Parameter File, so a commanded acquisition start after any manual commanding will use the last commanded STIM state, but after an instrument reset the EEPROM value will be used..			
	ident	**size**	**description**	
Parameters:	none	-	-	
Acceptance:	always			
Execution:	when hardware control is successfully deactivated			
Errors:	0x88 ecSTIM_NOT_DEACTIVATED			

Mnemonic:	LADISCSET		
Command code:	0x660C	**Critical Command:**	No
Command size:	12 bytes	**Disabled in SAFE state:**	No
Description:	Set discriminator to the specified value. Executing this command will also set the entry 'P_DISCRIMINATOR' in the active copy of the Parameter File to the specified value, so a commanded acquisition start after any manual commanding will use the last commanded Discriminator level, but after an instrument reset the EEPROM value will be used.		
	ident	**size**	**description**
Parameters:	DISC_LEVEL -	1 bytes 3 bytes	Level (1 byte) unused
Acceptance:	always		
Execution:	always		
Errors:	none		

Mnemonic:	LAAPDORCLS		
Command code:	0x660D	**Critical Command:**	No
Command size:	8 bytes	**Disabled in SAFE state:**	No
Description:	Command the aperture door to the closed position. Aperture door is always commanded to the requested close position for the full period specified in the parameter value (P_DOOR_CONTROL). At the end of the control period either execution completion or when the door sensors do not report closed state an error is reported.		
	ident	**size**	**description**
Parameters:	none	-	-
Acceptance:	Always, immediate after command processing started.		
Execution:	When door status reports closed position at end of control period.		
Errors:	0x48 ecCOMMANDED_DOOR_MOVE_FAILED		

Mnemonic:	LAHVPSDEACT		
Command code:	0x660E	**Critical Command:**	No
Command size:	8 bytes	**Disabled in SAFE state:**	No
Description:	Command HVPS off, first change the HV setpoint to zero and then disable the HVPS control. The command itself never fails but an error message may result when this command terminates a previous HVPS activation (rampup) command. Executing this command will also set the entry P_HV_LEVEL in the active copy of the Parameter File to zero, so a commanded acquisition start after any manual commanding will use the last commanded HV level, but after a reset the EEPROM value will again be used.		
	ident	**size**	**description**
Parameters:	none	-	-
Acceptance:	always		
Execution:	always		
Errors:	0x83 ecHV_RAMPUP_TERMINATED		

Mnemonic:	LAAPDOROPN		
Command code:	0x660F	**Critical Command:**	Yes
Command size:	8 bytes	**Disabled in SAFE state:**	Yes

Mnemonic:	LAAPDOROPN		
Description:	Command the aperture door to the open position. Aperture door is commanded to the requested open position for the full period specified in the parameter value (P_DOOR_CONTROL). At the end of the control period either execution completion or when the door sensors do not report open state an error is reported.		
	ident	**size**	**description**
Parameters:	none	-	-
Acceptance:	Direct; command is stored when waiting for 'Confirm Critical'. 'Confirm Critical' is accepted when the instrument is in CHECKOUT state.		
Execution:	After correct confirmation, when door status reports closed position at end of control period.		
Errors:	0x48 ecCOMMANDED_DOOR_MOVE_FAILED		

Mnemonic:	LAHVPSACT		
Command code:	0x6610	**Critical Command:**	Yes
Command size:	12 bytes	**Disabled in SAFE state:**	Yes
Description:	Command the HVPS to the specified level. The HV is commanded to follow a slow rampup to prevent damage to the detector. The rampup timing is controlled by parameter file values (P_HV_STEP_SIZE and P_HV_STEP_TIME). Decreases to a lower HV level are executed immediately. Steps to a higher HV level are executed using a fraction of the remaining difference between current value and commanded value. This means that initially larger steps will be taken but at the end of the ramp-up small steps are used. After each step the ramp up waits the specified time (P_HV_STEP_TIME) to let the HV settle. During the ramp-up the safety checking is active (as always) and the HV parameters will be checked according to the safety checks every 100 ms. Executing this command will also set the entry P_HV_LEVEL in the active copy of the Parameter File to the specified level, so a commanded acquisition start after any manual commanding will use the last commanded HV level, but after a reset the EEPROM value will be used again. Primary and redundant HV power supplies are connected only to the corresponding power busses, so a HVPS can only be activated when the corresponding power bu sis active. In addition the parameter file provides also a software based mechanism to select either HVPS.		
	ident	**size**	**description**
Parameters:	HV_LEVEL	1 byte	HV Level (range 0 – 214, 0 – 6.0 kVolt, 175 ~ 4.95 kVolt)
	unused	3 bytes	-
Acceptance:	Direct; command is stored when waiting for 'Confirm Critical'. 'Confirm Critical' is accepted when the instrument is in CHECKOUT state and commanded value is below the maximum allowed command value as defined in the parameter file (P_HV_MAX_HVSET).		
Execution:	After correct confirmation and checks, when requested HV level is reached.		
Errors:	0x80 ecHV_COMMANDED_ABOVE_MAXIMUM_LIMIT 0x81 ecHVPS_OFF_BEFORE_RAMPUP_COMPLETE 0x82 ecHV_LEVEL_SET_ABOVE_MAXIMUM_LIMIT 0x83 ecHV_RAMPUP_TERMINATED		

Mnemonic:	LAHTRCTRL		
Command code:	0x6611	**Critical Command:**	No
Command size:	12 bytes	**Disabled in SAFE state:**	Yes

Description:	Specify a temperature control set point for the control algorithm of the selected heater. Commanding the heater to a set point of zero will turn the heater off, commanding a heater to the maximum level (255) will turn the heater on continuously. In all other cases the heater(s) will be activated whenever the measured temperature is below the set point. Heaters will immediately be turned off whenever the instrument transitions to the SAFE state. The heater control algorithm evaluates the set point and measured temperature every 10 seconds, so it may take up to 10 seconds before a specified heater is activated after the command is send and the measured temperature is below the set point. The parameter file entries ‘P_HtrSenseMirror‘ and ‘P_HtrSenseGrating’ control whether either the primary or the redundant temperature sensor is used by the temperature control algorithm. The parameter file entries ‘P_MirrorHtr?Enable‘ and ‘P_GratingHtr?Enable’ control which of the two redundant heaters may be activated, these parameters may also select both or neither heaters of a pair. Heaters are only connected to the corresponding power bus so redundant heaters can only be activated when the corresponding power bus is activated.		
	ident	**size**	**description**
Parameters:	HEATER	1 byte	Heater id 1 – Mirror decontamination heater 2 – Grating decontamination heater
	SETPOINT	1 byte	Heater set point (0-255, -30 - +60 ˚C)
	unused	2 bytes	-
Acceptance:	When the instrument is in CHECKOUT state and a correct heater is specified.		
Execution:	Direct when the requested set point is loaded in the temperature controller for the specified heater.		
Errors:	0x90 ecINCORRECT_HEATER_IDENTIFICATION		

Mnemonic:	LAWPASMAACT		
Command code:	0x6612	**Critical Command:**	Yes
Command size:	12 bytes	**Disabled in SAFE state:**	Yes

Description: Activate the primary, the redundant or both actuators of a given actuator pair. The activation of any given actuator control will be terminated if:

- the actuation period times out,
- in case of the WPA the WPA switch is activated and the switch control is enabled (P_WpaSensorEnable)
- the system changes to the SAFE state,
- a deactivation WPA/SMA command is sent,

Duration of the actuator activation for actuators labeled as SMA is defined by the parameter file entry P_TINI_CONTROL (in a range from 5 ms – 1.275 sec, with a typical value of 70 ms for the real SMAs, but dependant on temperature and typical 5 ms for the vent valve solenoid). Maximum duration of the actuator activation for the actuators labeled as WPA (detector door) is defined by the parameter file entry P_WPA_TIMEOUT (in a range 10 sec – 2550 sec, a typical activation takes 150 seconds but is dependant on temperature).
Primary and redundant actuators are only connected to the corresponding power busses, so they can only be activated when the corresponding bus is active.

	ident	**size**	**description**
Parameters:	ACTUATOR	1 byte	Actuator id 0 – none, deactivate all 1 – primary actuator door lock actuator (SMA) 2 – redundant actuator door lock actuator (SMA) 3 – both actuator door lock actuators (SMA) 4 – vent valve open (solenoid (SMA)) 5 – vent valve close (solenoid (SMA)) 6 – Not to be used (both vent valve coils) 7 – primary failsafe door (SMA) 8 – redundant failsafe door (SMA) 9 – both failsafe door actuators (SMA) 10 – primary detector door actuator (WPA) 11 – secondary detector door actuator (WPA) 12 – both detector door actuators (WPA)
	unused	3 bytes	-

Acceptance: Direct; command is stored when waiting for ‘Confirm Critical’. ‘Confirm Critical’ is accepted when the instrument is in CHECKOUT state and:

- The controlling task is not active and a valid actuator is specified or
- The actuator is commanded off

Execution: After correct confirmation, when execution completes without being terminated

Errors:

0x40	ecINCORRECT_ACTUATOR_IDENT_NUMBER
0x41	ecACTUATOR_CONTROLLER_TASK_BUSY
0x42	ecSMA_CONTROL_TERMINATED
0x43	ecWPA_CONTROL_TERMINATED
0x44	ecSMA_ZERO_DURATION_ACTIVATION
0x45	ecWPA_ZERO_DURATION_ACTIVATION
0x46	ecWPA_SWITCH_ENABLED_AND_ACTIVE

Mnemonic:	LASELFTEST			
Command code:	0x6613		**Critical Command:**	No
Command size:	12 bytes		**Disabled in SAFE state:**	Yes
Description:	Perform the self test sequences specified in the test mask. The duration of the command execution depends on the selected tests. The tests are executed starting by the test selected by the least significant bit to the test selected by the most significant bit. At the start of the test execution the reported test status byte (Hk_TEST_STATUS) will be set to 0xff and this status will be reported in at least one HK packet.. Whenever one of the specified tests completes successfully the corresponding bit will be cleared. Bits corresponding to not selected test will always be cleared (in order of test selection).			
	ident	**size**	**description**	
Parameters:	TEST_MASK	1 byte	Test mask (1 byte) 0x01 – Acquisition memory A (34 sec) 0x02 – Acquisition memory B (32 sec) 0x04 – Acquisition page switching (33 sec) 0x08 – unused Data RAM (53 sec) 0x10 – Code Memory (EXEC) (29 sec) 0x20 – EEPROM Parameter files (<1 sec) 0x40 – EEPROM pages 1, 3 (52 sec) 0x80 – EEPROM pages 2, 4 (52 sec)	
	unused	3 bytes		
Acceptance:	When the instrument is in CHECKOUT state and the ‘slow’ Task is ready to accept a command.			
Execution:	When execution of the selected tests by the ‘slow’ Task completes			
Errors:	0xd0 ecNO_TEST_START_SLOW_TASK_BUSY 0xd1 ecEEPROM1_CHECKSUM_MISMATCH 0xd3 ecEEPROM2_CHECKSUM_MISMATCH 0xd3 ecEEPROM3_CHECKSUM_MISMATCH 0xd4 ecEEPROM4_CHECKSUM_MISMATCH			

Mnemonic:	LAPRGSTRT		
Command code:	0x6614	**Critical Command:**	Yes
Command size:	12 bytes	**Disabled in SAFE state:**	Yes

Description: (Re)start execution of the onboard software at specified address. The three planned reasons for execution of this command are:

1- Start execution of the code in the PROM memory, this corresponds to a software reset and causes a complete reinitialization of the software environment, corresponding to startup after completion of the system hardware reset,

2- Start execution of the code in one of the four EEPROM pages upon TC command, after an EEPROM page has been loaded with updated code, or with the initial stored copy of the code in PROM,

3- Start execution of some special code entry to perform a function that is not supported by the standard commands, like executing the duplicate PROM code or executing a special test function that has previously been uploaded into RAM or EEPROM.

Note that in the processor mapping the PROM and EEPROM memory appears in the bottom 32 kbyte of the memory space, the top 32 kbyte always maps to the shared (code and data space) RAM memory. This RAM memory may still be used to contain executable code as long as it doesn't interfere with any general data variables used by the software.

	ident	**size**	**description**
Parameters:	CODE_MEMORY	1 byte	0x51 – EEPROM page 1 0x52 – EEPROM page 2 0x53 – EEPROM page 3 0x54 – EEPROM page 4 0x56 – PROM memory
	unused	1 byte	-
	START_ADDRESS	2 bytes	Start address (address range 0..65535)

Acceptance: Direct; command is stored when waiting for 'Confirm Critical'. 'Confirm Critical' is accepted when the instrument is in CHECKOUT state and a correct memory type is specified for the commanded restart. This last acceptance may not be reported though as the restart is executed immediately and only if the called function returns the acceptance reporting can be performed.

Execution: After correct confirmation and parameter verification, always. The execution (status) may not be reported though, as the restart is executed immediately and only if the called function returns the execution reporting can be performed.

Errors: 0x99 ecINCORRECT_MEMORY_TYPE_FOR_RESTART

Mnemonic:	LAREQTURNOFF		
Command code:	0x6615	**Critical Command:**	Yes
Command size:	12 bytes	**Disabled in SAFE state:**	Yes
Description:	Set request power down flag in the LAMP HK packets. In response the spacecraft will power down the LAMP instrument. The request flag remains set until another request is recieved by the instrument (or the instrument is reset). This command is only meant to support testing of the instrument request system, other (spacecraft) command options would be available to directly command a power down or instrument reset.		
	ident	**size**	**description**
Parameters:	REQUEST	1 byte	0 – no request 1 – request power off
	unused	3 bytes	-
Acceptance:	Direct; command is stored when waiting for 'Confirm Critical'. 'Confirm Critical' is accepted when the instrument is in CHECKOUT state and a valid request is specified.		
Execution:	After correct confirmation, always.		
Errors:	0x98	ecINCORRECT_REQUEST_STARTMASK	

Mnemonic:	LARSTTCSTAT		
Command code:	0x6616	**Critical Command:**	No
Command size:	8 bytes	**Disabled in SAFE state:**	Yes
Description:	Reset TC failure status detection (start receiving from both TC channels again). Execution of this command will also reset the 'last failed command' and 'last failure code' fields that are reported in the housekeeping packet.		
	ident	**size**	**description**
Parameters:	none		
Acceptance:	When the instrument is in CHECKOUT state.		
Execution:	always		
Errors:	none		

Mnemonic:	LAMEMCHK		
Command code:	0x6617	**Critical Command:**	No
Command size:	16 bytes	**Disabled in SAFE state:**	Yes

Description: Calculate a 16-bit byte wise xor rotate checksum over a specified block of memory, the calculated value is reported in the standard HK packet.
At the start of the execution of this command the memory checksum value reported in the HK data packet will be set to zero.
The memory checksum value reported in the HK data packet will contain the valid result when the completion of command execution is indicated. Calculation of the checksum is dependant on the block size, over a 32 kbyte of CODE (PROM) or EEPROM space will take about 24 seconds.

Parameters:

ident	size	description
START_ADDRESS	4 bytes	Start address
LENGTH	2 bytes	Number of bytes
MEMORY_TYPE	1 byte	Memory type from where the data should be checked, for P-LAMP the following types are defined for dump operations: 0x50 – DATA memory (addr range 0..65535) 0x51 – EEPROM page 1 (addr range 0..32767) 0x52 – EEPROM page 2 (addr range 0..32767) 0x53 – EEPROM page 3 (addr range 0..32767) 0x54 – EEPROM page 4 (addr range 0..32767) 0x55 – Acquisition memory (addr range 0.. 65535, by definition inactive page) 0x56 – CODE memory (address range 0..65535)
unused	1 byte	

Acceptance: When the instrument is in CHECKOUT state, a correct memory specification is received and the ‘slow’ Task is available for command execution.

Execution: After completion of the checksum calculation always

Errors:

0x60	ecDATAMEM_START_TOO_LARGE
0x61	ecDATAMEM_LENGTH_TOO_LARGE
0x62	ecDATAMEM_INCORRECT_BLOCK_SIZE
0x63	ecACQMEM_START_TOO_LARGE
0x64	ecACQMEM_LENGTH_TOO_LARGE
0x65	ecACQMEM_INCORRECT_BLOCK_SIZE
0x66	ecCODEMEM_START_TOO_LARGE
0x67	ecCODEMEM_LENGTH_TOO_LARGE
0x68	ecCODEMEM_INCORRECT_BLOCK_SIZE
0x69	ecEEPROM_MEM_START_TOO_LARGE
0x6a	ecEEPROM_MEM_LENGTH_TOO_LARGE
0x6b	ecEEPROM_MEM_INCORRECT_BLOCK_SIZE
0x6c	ecMEMORY_INCORRECT_MEMORY_TYPE
0x6d	ecZERO_LENGTH_MEM_BLOCK
0x70	ecNO_MEMCHECK_START_SLOW_TASK_BUSY

Mnemonic:	LAMEMLD		
Command code:	0x6618	**Critical Command:**	Yes
Command size:	20-144 bytes	**Disabled in SAFE state:**	Yes

Description: Load the specified memory with new data/code, up to 128 bytes can be loaded with a single command.
When loading into EEPROM the block may not straddle a 128-byte boundary and code must not be executed from same EEPROM page.
Verification of the memory load includes a comparison between the data read back from the updated memory and the data that was loaded into the memory. This may fail if the memory written to is not actual memory but control registers that are not read/write accessible as are present in the DATA memory space (see C&DH memory map in Appendix A)

Parameters:

ident	size	description
START_ADDRESS	4 bytes	Start address for the data to be written, specified within the instrument address space.
LENGTH	2 bytes	Number of bytes to be loaded, maximum of 128.
MEMORY_TYPE	1 byte	Memory type where the data should be loaded, for P-LAMP the following types are defined for load operations: 0x50 – DATA memory (address range 0..65535) 0x51 – EEPROM page 1 (addr range 0..32767) 0x52 – EEPROM page 2 (addr range 0..32767) 0x53 – EEPROM page 3 (addr range 0..32767) 0x54 – EEPROM page 4 (addr range 0..32767) 0x55 – Acquisition memory (addr range 0..65535, by definition inactive page)
unused	1 byte	-
DATA	n bytes	Data to be loaded into memory (1-128 bytes).
pad	f bytes	Pad up to size that is a multiple of 4 (0-3 bytes).

Acceptance: Direct; command is stored when waiting for ‘Confirm Critical’. ‘Confirm Critical’ is accepted when the instrument is in CHECKOUT state, a correct memory specification is received and the ‘slow’ Task is available for command execution.

Execution: After correct confirmation and parameter verification when the successful load of the data is verified

Errors:

0x60	ecDATAMEM_START_TOO_LARGE
0x61	ecDATAMEM_LENGTH_TOO_LARGE
0x62	ecDATAMEM_INCORRECT_BLOCK_SIZE
0x63	ecACQMEM_START_TOO_LARGE
0x64	ecACQMEM_LENGTH_TOO_LARGE
0x65	ecACQMEM_INCORRECT_BLOCK_SIZE
0x69	ecEEPROM_MEM_START_TOO_LARGE
0x6a	ecEEPROM_MEM_LENGTH_TOO_LARGE
0x6b	ecEEPROM_MEM_INCORRECT_BLOCK_SIZE
0x6c	ecMEMORY_INCORRECT_MEMORY_TYPE
0x6d	ecZERO_LENGTH_MEM_BLOCK
0x73	ecNO_MEMLOAD_LENGTH_ABOVE_128
0x74	ecNO_MEMLOAD_INTO_CODE_PROM
0x75	ecNO_EEPROM_LOAD_EXECUTING_EEPROM
0x76	ecSLOW_TASK_BUSY
0x77	ecNO_MEMLOAD_BLOCK_STRADDLES_128

Mnemonic:	LAMEMDMP		
Command code:	0x6619	**Critical Command:**	No
Command size:	20 bytes	**Disabled in SAFE state:**	Yes

Description: Generate memory dump packages on the low rate telemetry interface, multiple dump packages may be generated when the request it too large to be transferred in a single CCSDS memory dump packet (maximum P-LAMP dump package transfers 128 data bytes).
The duration of the command execution is influenced by the memory dump flow control that is executed by the spacecraft (flow control is included in the time synchronization message). Generation of memory dump packets may be prevented for an arbitrary period of time.
An ongoing memory dump command will be terminated when the instrument leaves the CHECKOUT state.

	ident	size	description
Parameters:	START_ADDRESS	4 bytes	Start address for the data to be read, specified within the instrument address space.
	LENGTH	4 bytes	Number of bytes to be dumped
	MEMORY_TYPE	1 byte	Memory type from where the data should be dumped, for P-LAMP the following types are defined for dump operations: 0x50 – DATA memory (addr range 0..65535) 0x51 – EEPROM page 1 (addr range 0..32767) 0x52 – EEPROM page 2 (addr range 0..32767) 0x53 – EEPROM page 3 (addr range 0..32767) 0x54 – EEPROM page 4 (addr range 0..32767) 0x55 – Acquisition memory (addr range 0.. 65535, by definition inactive page) 0x56 – CODE memory (address range 0..65535)
	unused	3 bytes	-

Acceptance: When the instrument is in CHECKOUT state, a correct memory specification is received and the ‘slow’ Task is available for command execution.

Execution: After completion of the dump operation (duration also depending on spacecraft flow control) always.

Errors:

0x60	ecDATAMEM_START_TOO_LARGE
0x61	ecDATAMEM_LENGTH_TOO_LARGE
0x62	ecDATAMEM_INCORRECT_BLOCK_SIZE
0x63	ecACQMEM_START_TOO_LARGE
0x64	ecACQMEM_LENGTH_TOO_LARGE
0x65	ecACQMEM_INCORRECT_BLOCK_SIZE
0x66	ecCODEMEM_START_TOO_LARGE
0x67	ecCODEMEM_LENGTH_TOO_LARGE
0x68	ecCODEMEM_INCORRECT_BLOCK_SIZE
0x69	ecEEPROM_MEM_START_TOO_LARGE
0x6a	ecEEPROM_MEM_LENGTH_TOO_LARGE
0x6b	ecEEPROM_MEM_INCORRECT_BLOCK_SIZE
0x6c	ecMEMORY_INCORRECT_MEMORY_TYPE
0x6d	ecZERO_LENGTH_MEM_BLOCK
0x71	ecNO_MEMDUMP_START_SLOW_TASK_BUSY
0x72	ecMEMDUMP_HALTED_NO_CHECKOUT_STATE
0x79	ecMEMORY_DUMP_TERMINATED

Appendix G - Internally Stored Parameter List

A number of instrument operations are controlled by configuration parameters stored in a table called the Parameter File. The current contents of this table in data memory (RAM) can be changed using the 'LAMP_SET_PARAMETER' command and the current contents can be inspected using the parameter reporting function in the HK packet. Upon system startup the Parameter File is initialized with values read from a triple redundant store in non-volatile (EEPROM) memory. If this initial restore fails the value of the specific parameter defaults to a hardcoded backup value. The list with default values specify the parameter file of the FM status on 2004/09/02. The names of the parameter fields (all names with the prefix 'P_') match the parameter names (without the 'P_') for the 'LAMP_SET_PARAMETER' command used to modify the corresponding field.

Table 30: LRO-LAMP internally stored parameter list

#	Parameter (size in bits where applicable)	Size in bytes	Description	Default value	Hardcoded (backup) value
	General control parameters				
0	Spare (1)	1	-	0	0
	P_HvpsAuto		LTS based control of HVPSr	1	1
	P_DoorAuto		LTS based control of apdoor	0	0
	P_DoorEnable (1)		Enable door close on safety (1 = close on safety)	1	1
	Spare (1)		-	0	0
	P_WpaSensorEnable (1)		Enable wax pellet actuator sensor feedback (1 = enabled)	1	1
	P_TcDisable (2)		0 – both enabled; 1,3 – disable channel 1; 2 disable channel 2	0	0
1	Spare (1)	1	-	0	0
	P_HtrSenseGrating (1)		Grating heater sensor select; 0=primary, 1=secondary	0	0
	P_GratingHtr1Enable (1)		Grating heater 1 control enabled; 1=enabled	1	1
	P_GratingHtr2Enable (1)		Grating heater 2 control enabled; 1=enabled	1	1
	Spare (1)		-	0	0
	P_HtrSenseMirror (1)		Mirror heater sensor select; 0=primary, 1=secondary	0	0
	P_MirrorHtr1Enable (1)		Mirror heater 1 control enabled; 1=enabled	1	1
	P_MirrorHtr2Enable (1)		Mirror heater 2 control enabled; 1=enabled	1	1
2	P_CriticalTimeout	1	Critical command timeout period in seconds, note the minimal used value of this parameter is automatically limited to 5 seconds to prevent a situation where the timeout is set so short that no parameter modifications can be made anymore	30	30

#	Parameter (size in bits where applicable)	Size in bytes	Description	Default value	Hardcoded (backup) value
3	P_TcMaxError	1	Number of errors allowed on any TC channels before disabling channel (0 – not active)	5	5
4	P_WpaTimeout	1	Wax pellet actuator timeout in seconds*10 (0-2550 sec), after the specified time the actuator will be disabled	20 (200 sec)	20 (200 sec)
5	P_TiNiControl	1	Shape metal actuators control time in seconds/200 (0-1.275 sec), shape metal actuator signals will be active for the specified amount of time without feedback	14 (70 ms)	14 (70 ms)
6	P_DoorControl	1	Aperture door motion control time in seconds/10 (0-25.5 sec), door control will be active for the specified amount of time without feedback	18 (1.8 sec)	18 (1.8 sec)
7	P_ReportParam	1	Current parameter value reported in housekeeping (255 results in cyclic reporting).	255 (cyclic)	255 (cyclic)
8	P_ReportSample	1	Number of sub-sample reporting cycles for the parameter reporting process.	1	1
9	Spare (4)	1	-		
	P_HW version id (4)		H/W Board Version Id included in HK TM packet to identify which model generated the received/recorded HK TM packet.	3 (FM)	7 (PROM default)
	Acquisition control parameters				
10	Spare (7)	1	-	0	0
	P_StimEnable		Enable Pixel STIM at start of acquisition	1	1
11	Spare (6)	1	-	0	0
	P_Hvps1Enable (1)		Enable Primary HVPS when commanded (1 = enabled)	1	0
	P_Hvps2Enable (1)		Enable Secondary HVPS when commanded (1 = enabled)	1	0
12	P_Discriminator	1	Discriminator set level (units, DAC counts)	89 (~1.05 Volt)	43 (~0.5 Volt)
13	P_HvLevel	1	High voltage operating level (units, DAC counts)	175 (~4.95 kV)	175 (~4.95 kV)
14	P_HvStepSize	1	High voltage step size (units, DAC counts)	55 (16/55 = 0.29)	55 (16/55 = 0.29)
15	P_HvStepTime	1	High voltage step duration in seconds (0-255)	6	6

#	Parameter (size in bits where applicable)	Size in bytes	Description	Default value	Hardcoded (backup) value
16	P_HvSafeLevel	1	Safe High voltage operating level when limited by LTS (units, DAC counts)	118 (~3.5 kV)	118 (~3.5 kV)
17	P_PixelListHack	1	Time hack used for pixel list acquisitions (0- 1/256, 1-1/128, 2-1/64, 3-1/32, 4-1/16, 5-1/8, 6-1/4, 7-1/2)	0 (4 ms)	0 (4 ms)
18, 19	P_HistoExpDuration	2	Duration of one histogram acquisition exposure in seconds (1-65535)	100	100
20, 21	P_AcqTimeout	2	Acquisition timeout, defines backup acquisition termination, specifies maximum acquisition duration in seconds (1-65535)	7000	7000
22	P_HotSeg1	1	Hot segment 1; Each hot segment specification masks out detector events in a rectangular area (32 spectral * 4 spatial pixels), top three bits specify the spatial position of this window, remaining 5 bits specify the spectral position of the window (see section 2.1.1.4).	0	0
23	P_HotSeg2	1	Hot segment 2	0	0
24	P_HotSeg3	1	Hot segment 3	0	0
25	P_HotSeg4	1	Hot segment 4	0	0
26	P_HotSeg5	1	Hot segment 5	0	0
27	P_HotSeg6	1	Hot segment 6	0	0
28	P_HotSeg7	1	Hot segment 7	0	0
29	P_HotSeg8	1	Hot segment 8	0	0
	LTS control parameters				
30, 31	P_LtsAOffset	2	Offset for LTS sensor A adjustment	0	0
32, 33	P_LtsBOffset	2	Offset for LTS sensor B adjustment	0	0
34	Spare (4)	1	-	0	0
	P_LtsAGain (4)		Gain for LTS sensor A adjustment	6 (/64)	6 (/64)
35	Spare (4)	1	-	0	0
	P_LtsBGain (4)		Gain for LTS sensor B adjustment	6 (/64)	6 (/64)
36	P_LtsADark	1	Threshold for LTS sensor A on	64	64
37	P_LtsBDark	1	Threshold for LTS sensor B on	64	192
38	P_LtsALight	1	Threshold for LTS sensor A off	192	64
39	P_LtsBLight	1	Threshold for LTS sensor B off	192	192
40	P_LtsDarkMode (4)	1	Logic Mode selection function LTS-Dark On control	0x8 (both)	0x8 (both)
	P_LtsLightMode (4)		Logic Mode selection function LTS Dark Off control	0xe (either)	0xe (either)

#	Parameter (size in bits where applicable)	Size in bytes	Description	Default value	Hardcoded (backup) value
41	P_LtsDarkDelay	1	Delay for LTS-Dark signal change to ‘on’ in units of 100 ms (0-25.5 sec)	45 (4.5 sec)	45 (4.5 sec)
42	P_LtsLightDelay	1	Delay for LTS -Dark signal change to ‘off’ in units of 100 ms (0-25.5 sec)	5 (0.5 sec)	5 (0.5 sec)
	Safety checking parameters				
43	P_LtsMaxCycles	1	Maximum number of LTS induced HV cycles per acquisition	12	12
44, 45	P_MaxCountRate	2	Maximum allowed count rate (Hz)	15000	15000
46	P_HvLowSafety	1	HV lowest voltage setting above which the safety checking can be activated, this prevents problems with lower region of control not being linear etc.	58 (~2.0 kV)	58 (~2.0 kV)
47	P_DacAdcFactor	1	Conversion from to DAC setting to ADC read back used in HVPS checking (factor/240)	208 (0.87)	208 (0.87)
48	P_HvMaxHvSet	1	Maximum allowed HV setpoint voltage (DAC)	184 (~5.2 kV)	184 (~5.2 kV)
49	P_HvMcpTol	1	MCP voltage tolerance (units, ADC counts)	4 (~100 V)	4 (~100 V)
50	P_HvFailMcp	1	MCP voltage max fail count	5	5
51	P_HvMaxStripI	1	Maximum allowed strip current (ADC counts)	188 (~20 uA)	188 (~20 uA)
52	P_HvFailStrip	1	Strip current max fail count	5	5
53	P_HvMinAnodeV	1	Minimum allowed anode voltage (ADC counts)	186 (~ 589 V)	186 (~ 589 V)
54	P_HvMaxAnodeV	1	Maximum allowed anode voltage (ADC counts)	199 (~ 631 V)	199 (~ 630 V)
55	P_HvFailAnode	1	Anode voltage max fail count	5	5
56	P_MaxMir1Temp	1	Maximum allowed temperature (ADC counts)	220 (~ 50 ˚C)	220 (~ 50 ˚C)
57	P_MaxMir2Temp	1	Maximum allowed temperature (ADC counts)	220 (~ 50 ˚C)	220 (~ 50 ˚C)
58	P_MaxGrat1Temp	1	Maximum allowed temperature (ADC counts)	215 (~ 45 ˚C)	215 (~ 45 ˚C)
59	P_MaxGrat2Temp	1	Maximum allowed temperature (ADC counts)	215 (~ 45 ˚C)	215 (~ 45 ˚C)
60	P_MaxCdhTemp	1	Maximum allowed temperature (ADC counts)	224 (~ 55 ˚C)	224 (~ 55 ˚C)
61	P_MaxDetHTemp	1	Maximum allowed temperature (ADC counts)	215 (~ 45 ˚C)	215 (~ 45 ˚C)

#	Parameter (size in bits where applicable)	Size in bytes	Description	Default value	Hardcoded (backup) value
62	P_TempMask	1	Mask to disable certain temperature sensors from the temperature safety check (1=masked, 0- enabled), the 6 least significant bits in the mask correspond to the 6 temperature sensors listed above in the same order.	0x00	0x00
63	P_SafetyMask	1	Initial startup value for the safety mask and override (1=masked, 0- enabled) - Override (1-all checks masked, 0x80) - Spare (1) - Temperature safety (0x20) - Cycle safety (0x10) - Anode safety (0x08) - Strip safety (0x04) - HV safety (0x02) - Bright safety (0x01)	0x00	0x00
64, 65	P_SafetyTimeout	2	Safety timeout (seconds)	60	60
	Special / Test parameters				
66	P_DebugTest	1	Debug / Test setting	2	2
67,68	P_Spare_1	2	Unassigned	0	0
69,70	P_Number of modifications	2	Accumulated count of changes made to the parameter memory in EEPROM memory to provide a tracking mechanism for the maximum number of writes cycles.	0	0
71 -	P_Spare_2	57	Pad size of table to a full EEPROM block	0	0
	Total table size in bytes	128	Table stored redundantly in internal non-volatile instrument memory		

Appendix H - Error Codes

Table 31: LAMP Error Codes

LRO-LAMP: ErrorDef.h, version: 1.3 - Date: 2006/10/11 16:35:50

```
  ecUNUSED_00                        = 0x00,
/* ITF decoding errors --------------------------------------------------*/
/* _1 and _2 indicate channel on which byte was received                 */
ecINCORRECT_ITF_CHECKSUM_1           = 0x01, // Tc::handleTcByte()
ecINCORRECT_ITF_CHECKSUM_2           = 0x02, // Tc::handleTcByte()
ecUNSUPPORTED_ITF_MESSAGE_TYPE_1     = 0x03, // Tc::handleTcByte()
ecUNSUPPORTED_ITF_MESSAGE_TYPE_2     = 0x04, // Tc::handleTcByte()
ecTOO_LARGE_ITF_CMD_PACKAGE_1        = 0x05, // Tc::handleTcByte()
ecTOO_LARGE_ITF_CMD_PACKAGE_2        = 0x06, // Tc::handleTcByte()
ecTOO_SHORT_ITF_CMD_PACKAGE_1        = 0x07, // TmTc::timeoutTcSerial(()
ecTOO_SHORT_ITF_CMD_PACKAGE_2        = 0x08, // TmTc::timeoutTcSerial(()
ecINCORRECT_1ST_ITF_SYNC_BYTE_1      = 0x09, // Tc::handleTcByte()
ecINCORRECT_1ST_ITF_SYNC_BYTE_2      = 0x0a, // Tc::handleTcByte()
ecINCORRECT_2ND_ITF_SYNC_BYTE_1      = 0x0b, // Tc::handleTcByte()
ecINCORRECT_2ND_ITF_SYNC_BYTE_2      = 0x0c, // Tc::handleTcByte()
ecINCORRECT_3RD_ITF_SYNC_BYTE_1      = 0x0d, // Tc::handleTcByte()
ecINCORRECT_3RD_ITF_SYNC_BYTE_2      = 0x0e, // Tc::handleTcByte()
ecITF_ERROR_WAITING_1_2              = 0x10, // TmTc::reportItfError()
ecTC_BUFFER_OVERFLOW_1               = 0x11, // TmTc::isrTmTc()
ecTC_BUFFER_OVERFLOW_2               = 0x12, // TmTc::isrTmTc()
ecRX_FLUSH_FAILED_1                  = 0x13, // TmTc::isrTmTc()
ecRX_FLUSH_FAILED_2                  = 0x14, // TmTc::isrTmTc()
/* TM gneration errors --------------------------------------------------*/
ecTM_TRANSFER_MISSED                 = 0x18, // Tm::taskTM()
ecTM_TRANSFER_ERROR                  = 0x19, // TmTc::transmitFrame()
/* General command decoding ---------------------------------------------*/
ecIMPROPER_LENGTH_FOR_COMMAND        = 0x20, // Tc::processTc()
ecUNKNOWN_UNRECOGNIZED_TELECOMMAND   = 0x21, // Tc::tcUnknown()
ecWRONG_NUMBER_OF_COMMAND_WORDS      = 0x22, // Tc::processTc()
ecCOMMAND_NOT_IN_SAFE_STATE          = 0x23, // Tc::processTc()
ecALREADY_CRITICAL_COMMAND_PENDING   = 0x24, // Tc::processTc()
ecINCORRECT_CONFIRM_CRITICAL         = 0x25, // Tc::processTc()
ecMISSING_CONFIRM_CRITICAL           = 0x26, // Tc::processTc()
ecNO_CRITCAL_COMMAND_PENDING         = 0x27, // Tc::tcConfirm()
ecCRITICAL_COMMAND_TIMEOUT           = 0x28, // Tc::timeoutCriticalCommand()
ecTIME_MESSAGE_TOO_SHORT             = 0x2c, // Clock::processTimeMessage()
ecTIME_MESSAGE_TOO_LONG              = 0x2d, // Clock::processTimeMessage()
/* Telecommand Execution ------------------------------------------------*/
ecFAILED_TO_CHANGE_TO_CHECKOUT       = 0x30, // State::tcCheckout()
ecINCORRECT_ACTUATOR_IDENT_NUMBER    = 0x40, // WpaSma::tcActivate()
ecACTUATOR_CONTROLLER_TASK_BUSY      = 0x41, // WpaSma::tcActivate()
ecSMA_CONTROL_TERMINATED             = 0x42, // WpaSma::activateSma()
ecWPA_CONTROL_TERMINATED             = 0x43, // WpaSma::activateDetDoorWpa()
ecSMA_ZERO_DURATION_ACTIVATION       = 0x44, // WpaSma::activateSma()
ecWPA_ZERO_DURATION_ACTIVATION       = 0x45, // WpaSma::activateDetDoorWpa()
ecWPA_SWITCH_ENABLED_AND_ACTIVE      = 0x46, // WpaSma::activateDetDoorWpa()
ecCOMMANDED_DOOR_MOVE_FAILED         = 0x48, // Door::taskDoor()
ecNO_ACQ_START_SAFETY_PENDING        = 0x51, // Slow::commandAcquisition()
ecNO_ACQ_START_SLOW_TASK_BUSY        = 0x52, // Slow::commandAcquisition()
ecNO_ACQ_START_WRONG_TEST_PARAM      = 0x53, // Slow::commandAcquisition()
ecNO_ACQ_START_WRONG_DOOR_PARAM      = 0x54, // Slow::commandAcquisition()
```

LRO-LAMP: ErrorDef.h, version: 1.3 - Date: 2006/10/11 16:35:50

```
ecDATAMEM_START_TOO_LARGE          = 0x60, // Slow::checkGeneralMemParams(
ecDATAMEM_LENGTH_TOO_LARGE         = 0x61, // Slow::checkGeneralMemParams(
ecDATAMEM_INCORRECT_BLOCK_SIZE     = 0x62, // Slow::checkGeneralMemParams(
ecACQMEM_START_TOO_LARGE           = 0x63, // Slow::checkGeneralMemParams(
ecACQMEM_LENGTH_TOO_LARGE          = 0x64, // Slow::checkGeneralMemParams(
ecACQMEM_INCORRECT_BLOCK_SIZE      = 0x65, // Slow::checkGeneralMemParams(
ecCODEMEM_START_TOO_LARGE          = 0x66, // Slow::checkGeneralMemParams(
ecCODEMEM_LENGTH_TOO_LARGE         = 0x67, // Slow::checkGeneralMemParams(
ecCODEMEM_INCORRECT_BLOCK_SIZE     = 0x68, // Slow::checkGeneralMemParams(
ecEEPROM_MEM_START_TOO_LARGE       = 0x69, // Slow::checkGeneralMemParams(
ecEEPROM_MEM_LENGTH_TOO_LARGE      = 0x6a, // Slow::checkGeneralMemParams(
ecEEPROM_MEM_INCORRECT_BLOCK_SIZE  = 0x6b, // Slow::checkGeneralMemParams(
ecMEMORY_INCORRECT_MEMORY_TYPE     = 0x6c, // Slow::checkGeneralMemParams(
ecZERO_LENGTH_MEM_BLOCK            = 0x6d, // Slow::checkGeneralMemParams(
ecNO_MEMCHECK_START_SLOW_TASK_BUSY = 0x70, // Slow::tcMemCheck()
ecNO_MEMDUMP_START_SLOW_TASK_BUSY  = 0x71, // Slow::tcMemDump()
ecMEMDUMP_HALTED_NO_CHECKOUT_STATE = 0x72, // Tm::taskTM()
ecNO_MEMLOAD_LENGTH_ABOVE_128      = 0x73, // Slow::tcMemLoad()
ecNO_MEMLOAD_INTO_CODE_PROM        = 0x74, // Slow::tcMemLoad()
ecNO_EEPROM_LOAD_EXECUTING_EEPROM  = 0x75, // Slow::tcMemLoad()
ecSLOW_TASK_BUSY                   = 0x76, // Slow::activateSlowTask()
ecNO_MEMLOAD_BLOCK_STRADDLES_128   = 0x77, // Slow::tcMemLoad()
ecMEMORY_LOAD_READBACK_MISMATCH    = 0x78, // Memory::slowLoadMemory()
ecMEMORY_DUMP_TERMINATED           = 0x79, // Memory::slowDumpMemory()
ecHV_COMMANDED_ABOVE_MAXIMUM_LIMIT = 0x80, // Hvps::tcHvOn()
ecHVPS_OFF_BEFORE_RAMPUP_COMPLETE  = 0x81, // Hvps::commandHvOff()
ecHV_LEVEL_SET_ABOVE_MAXIMUM_LIMIT = 0x82, // Hvps::commandHvOn()
ecHV_RAMPUP_TERMINATED             = 0x83, // Hvps::taskHvps()
ecSTIM_NOT_DEACTIVATED             = 0x88, // Detector::tcStimOff()
ecSTIM_NOT_ACTIVATED               = 0x89, // Detector::tcStimOn()
ecINCORRECT_HEATER_IDENTIFICATION  = 0x90, // Heater::tcHeater()
ecINCORRECT_REQUEST_STARTMASK      = 0x98, // System::tcRequest()
ecINCORRECT_MEMORY_TYPE_FOR_RESTART = 0x99, // System::tcStartProg()

/* Acquisition Control -------------------------------------------------*/
ecSYNCED_HISTO_ACQUISITION_FAILED  = 0xa0, // Acquire::slowAcquireHistogra
ecSYNCED_PIXEL_ACQUISITION_FAILED  = 0xa1, // Acquire::slowAcquirePixelLis
ecCLEAR_HISTO_ACQUISITION_FAILED   = 0xa2, // Acquire::slowAcquireHistogra
ecCLEAR_PIXEL_ACQUISITION_FAILED   = 0xa3, // Acquire::slowAcquirePixelLis
ecSTART_HISTO_ACQUISITION_FAILED   = 0xa4, // Acquire::slowAcquireHistogra
ecSTART_PIXEL_ACQUISITION_FAILED   = 0xa5, // Acquire::slowAcquirePixelLis
ecACQUISITION_TIMEOUT_OCCURED      = 0xa6, // Acquire::timeoutAcquisition(

/* Parameter File ------------------------------------------------------*/
ecPARAMETER_INDEX_OUT_OF_RANGE     = 0xb0, // Param::tcSetParm()
ecPARAMETER_READBACK_FAILED        = 0xb1, // Param::tcSetParm()
ecSTORE_PARAMS_READBACK_FAILED_1   = 0xb2, // Param::tcStoreParms()
ecSTORE_PARAMS_READBACK_FAILED_2   = 0xb3, // Param::tcStoreParms()
ecSTORE_PARAMS_READBACK_FAILED_3   = 0xb4, // Param::tcStoreParms()
ecSTORE_PARAMS_ONLY_WHEN_EXEC_PROM = 0xb5, // Param::tcStoreParms()
ecINCORRECT_TABLE_FOR_PARAM_LOAD   = 0xb6, // Param::tcLoadParms()
ecLOAD_PARAM_TABLE_1_MISMATCH      = 0xb7, // Param::loadMajorityEepromPar
ecLOAD_PARAM_TABLE_2_MISMATCH      = 0xb8, // Param::loadMajorityEepromPar
ecLOAD_PARAM_TABLE_3_MISMATCH      = 0xb9, // Param::loadMajorityEepromPar
ecLOAD_PARAM_ALL_DIFFERENT         = 0xba, // Param::loadMajorityEepromPar
```

LRO-LAMP: ErrorDef.h, version: 1.3 - Date: 2006/10/11 16:35:50

```
/* Internal Errors ------------------------------------------------------*/
ecTOO_MANY_BYTES_FOR_EEPROM_WRITE   = 0xc0, // Memory::writeEeprom()
ecTOO_MANY_BYTES_FOR_EEPROM_READ    = 0xc1, // Memory::readEeprom()

/* Self Test ------------------------------------------------------------*/
ecNO_TEST_START_SLOW_TASK_BUSY      = 0xd0, // Slow::tcSelfTest()
ecEEPROM1_CHECKSUM_MISMATCH         = 0xd1, // Memory::testAllCodeMemory()
ecEEPROM2_CHECKSUM_MISMATCH         = 0xd3, // Memory::testAllCodeMemory()
ecEEPROM3_CHECKSUM_MISMATCH         = 0xd3, // Memory::testAllCodeMemory()
ecEEPROM4_CHECKSUM_MISMATCH         = 0xd4, // Memory::testAllCodeMemory()

/* No errors ------------------------------------------------------------*/
ecNEXT_ERROR_LOG_SLOT               = 0xf8, // Tm::logError()
ecNO_ERROR_RESET                    = 0xfd, // Tm::resetLastFailed()
ecNO_ERROR_INITIAL                  = 0xfe, // Tc::Init()
ecUNUSED_FF                         = 0xff
```

Appendix I - Parameter Conversions

All conversion for the LAMP instrument that converts from commanded and measured counts into engineering units are defined by polynomial equations of order five. Anode Voltage, Strip Current and Discriminator voltage are degenerate linear cases where only a liner equation is needed. The factors of the polynomial equations are listed in Table 32

Table 32: Polynomial conversion factors

Factor	Temp	Hvset	McpV1	AnodeV	SumStripI	StripI (single)	Discrim
	℃	kVolt	kVolt	Volt	μAmp	μAmp	Volt
x^0	-7.803000E+01	0.000000E+00	0.000000E+00	0.000000E+00	0.000000E+00	0.000000E+00	0.000000E+00
x^1	2.385000E+00	-6.169450E-02	-6.83613E-02	-3.168035E+00	9.304640E-02	1.182147E-01	1.176471E-02
x^2	-4.087000E-02	8.076830E-04	8.62698E-04	0.000000E+00	0.000000E+00	0.000000E+00	0.000000E+00
x^3	3.752000E-04	-8.328130E-06	-8.49849E-06	0.000000E+00	0.000000E+00	0.000000E+00	0.000000E+00
x^4	-1.601000E-06	4.091090E-08	3.95830E-08	0.000000E+00	0.000000E+00	0.000000E+00	0.000000E+00
x^5	2.594000E-09	-7.689560E-11	-7.11661E-11	0.000000E+00	0.000000E+00	0.000000E+00	0.000000E+00

Note that the measurements from the Strip Current 1 and Strip Current 2 measure a part of the total current and that the distribution of the current over the two measurements changes depending on the commanded HV, this means that the two separate reading are virtually useless and that in all cases the summed maximum strip current should be used as the total strip current measurement. When both HVPSs are active the SumStripI field and it's calibration provide the complete current reading, when only a single HVPS is active the single conversion factors should be used together with the corresponding current telemetry (Strip Current 1 or Strip Current 2).

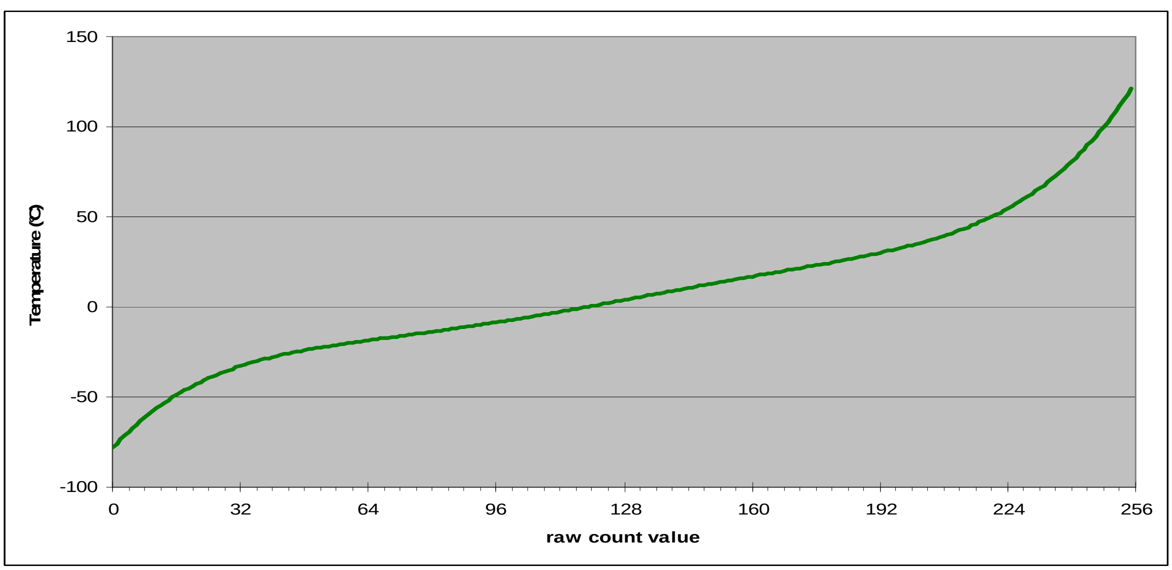

Figure 26: Temperature Conversion Curve

Table 33: Parameter Conversions Values

Counts	Temp	HvSet[1)]	McpV	AnodeV	SumStripI	StripI	Discrim
	deg C	kVolt	kVolt	Volt	uA	uA	V
0	-78.0	0.00	0.00	0	0.0	0.0	0.00
9	-59.6	-0.50	-0.55	-29	0.8	1.1	0.11
18	-46.3	-0.89	-1.00	-57	1.7	2.1	0.21
21	-42.8	**-1.01**	-1.13	-67	2.0	2.5	0.25
31	-33.6	-1.35	-1.51	-98	2.9	3.7	0.36
36	-30.2	**-1.50**	-1.68	-114	3.3	4.3	0.42
43	-26.3	-1.69	-1.90	-136	4.0	5.1	0.51
47	-24.5	-1.80	-2.01	-149	4.4	5.6	0.55
55	-21.4	**-2.00**	-2.24	-174	5.1	6.5	0.65
59	-20.1	-2.10	-2.35	-187	5.5	7.0	0.69
76	-15.0	**-2.51**	-2.80	-241	7.1	9.0	0.89
92	-10.0	-2.90	-3.24	-291	8.6	10.9	1.08
96	-8.6	**-3.00**	-3.35	-304	8.9	11.3	1.13
97	-8.3	-3.02	-3.38	-307	9.0	11.5	1.14
112	-2.7	-3.40	-3.80	-355	10.4	13.2	1.32
116	-1.1	**-3.50**	-3.91	-367	10.8	13.7	1.36
119	0.1	-3.57	-4.00	-377	11.1	14.1	1.40
122	1.4	-3.64	-4.09	-387	11.4	14.4	1.44
128	3.9	-3.79	-4.26	-406	11.9	15.1	1.51
137	7.6	**-4.01**	-4.52	-434	12.7	16.2	1.61
141	9.3	-4.10	-4.63	-447	13.1	16.7	1.66
143	10.1	-4.15	-4.69	-453	13.3	16.9	1.68
145	10.9	**-4.20**	-4.75	-459	13.5	17.1	1.71
149	12.5	-4.29	-4.86	-472	13.9	17.6	1.75
153	14.1	**-4.39**	**-4.98**	-485	**14.2**	18.1	1.80
158	16.1	-4.51	-5.12	-501	14.7	18.7	1.86
162	17.7	**-4.61**	-5.24	-513	15.1	19.2	1.91
166	19.2	-4.71	-5.36	-526	15.4	19.6	1.95
168	**20.0**	-4.76	-5.43	-532	15.6	19.9	1.98
170	**20.8**	**-4.81**	-5.49	-539	15.8	20.1	2.00
173	**21.9**	-4.89	-5.58	-548	16.1	20.5	2.04
175[1)]	**22.7**	**-4.95**	-5.65	-554	16.3	20.7	2.06
181	**25.1**	-5.12	-5.85	-573	16.8	21.4	2.13
185	26.8	-5.24	-5.99	-586	17.2	21.9	2.18
190	29.1	-5.41	-6.17	-602	17.7	22.5	2.24
192	30.0	-5.48	-6.25	**-608**	17.9	22.7	2.26
200	34.3	-5.81	-6.59	-634	18.6	23.6	2.35
209	40.2	-6.25	-7.04	-662	19.4	24.7	2.46
210	41.0	-6.30	-7.10	-665	19.5	24.8	2.47
220	50.1			-697	20.5	26.0	2.59
228	60.0			-722	21.2	27.0	2.68
255	121.3			-808	23.7	30.1	3.00

1) In bold the Nominal slow/test HV rampup sequence
2) In green nominal operating figures at room temperature

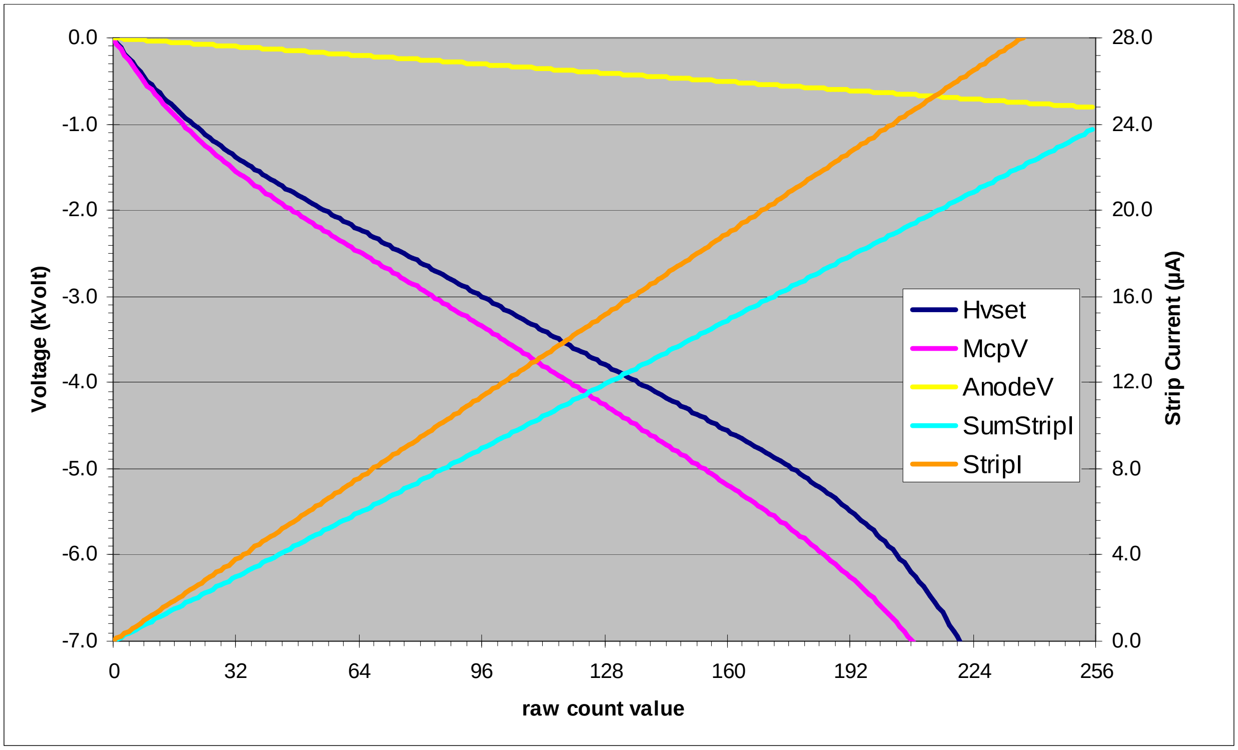

Figure 27: HVPS Parameters Conversion Curve

Science data:

Wavelength Calibration: wavelength (in nm) = `0.179 * column + 38.1`

Resulting in a detector range for the effective columns (150-900) of 65 to 199 nm

Appendix J - Differences between FM and LIISs

There are some differences between the LRO-LAMP Flight model instrument and the 3 LIISs (Engineering Models) that were manufactured during the LAMP project. An overview of the differences is presented in the following table:

Table 34: Unit differences

Parameter / Flight Unit	**LIIS 1**	**LIIS 2**	**LIIS 3**
Integrated on the Lunar Reconnaissance Orbiter spacecraft	Mini-LIIS, being rebuilt as the SwRI (software) support LIIS	Full LIIS, delivered to GSFC for I&T and mission support	C&DH only, delivered for GSFC for additional I&T support
Color of Box, covered in black MLI		Black anodized	Gray Box
Reported Housekeeping TM identifier (HW_VERSION) based on parameter configuration: 3 - FM	1 – L1	2 - L2	4 – L3
Soldered PROM with version 1.00 of software	Socketed EEPROM with version 1.00 of software	Socketed EEPROM with version 1.00 of software	Socketed EEPROM with version 1.00 of software

Appendix K - Mini LIIS Startup Instructions

The Mini LIIS is a limited version of the LAMP Instrument Interface Simulator, it contains An LVPS with only the A-side power supply installed and a complete LAMP C&DH. The installed software is version 0.01 of the LAMP Flight Software which provides a full TM and TC capability and an almost complete processing capability. Since some of the internal functions are missing some of the telemetry fields just provide static values.

When the power connector (see Appendix B) is connected to the instrument and the 31 Volt power is activated the power consumption is expected to be about 52 mA.
After about a 2-3 seconds internal startup and self-test the LIIS will start generating telemetry data on both the A and B side telemetry output. Data is generated RS-422 levels at 38k4 baud, one start bit, 8 data bits, no parity and one stop bit. Each second an ITF frame will be generated that contains a single housekeeping packet, the overall size of the generated frame is 132 bytes. An example of a generated frame is shown below, the detailed contents is described in Appendix D note that the exact contents of the housekeeping packet depends on external and configuration issues:

```
0x000: fe fa 30 04 6b 00 7d 00 00 00 08 81 c0 0c 00 73
0x010: 00 0f 42 4c 30 39 2d c0 30 01 00 00 00 00 00 00
0x020: ff ff fe 00 00 00 54 00 00 00 00 01 86 ca 10 40
0x030: 30 39 00 00 7f ff ff ff 00 00 00 02 00 00 02 00
0x040: 00 00 9f aa 1a 9d 16 1d 00 00 00 00 00 00 00 00
0x050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 57
0x060: 57 59 59 74 6c 10 3c 00 00 81 01 00 b8 02 0a 46
0x070: 00 1b 00 56 70 8a a4 be d8 f2 0c 26 40 a8 f2 20
0x080: 0d 25 3c b2
```

The integer part of the Spacecraft time that is reported in blue colored bytes in the TM frame will increment for each generated frame, after a reset the LAMP instrument will start counting the time at 1000000 (0x000F4240). The top nibble of the yellow byte contains the instrument state information, the value 2 indicates the SAFE state, other states are 3- CHECKOUT and 1- ACQUIRE. The bytes in green indicate the accepted command counter, after a reset the reported value will initially be zero.
The RS-422 compatible telecommand signal can either be connected to the A-side or B-side TC input. The serial data format is identical to the TM format: RS-422 levels at 38k4 baud, one start bit, 8 data bits, no parity and one stop bit. After issuing the LAMP_ENTER_CHECKOUT state telecommand the reported instrument state will change to CHECKOUT and the accepted command counter will be incremented by one. If the command is not accepted by LAMP the state will not change but the rejected command counter (located in two bytes after the accepted command counter) will be incremented, the red colored byte that marks the error code may indicate the reason why the command was not accepted (see appendix H):

```
LAMP_ENTER_CHECKOUT::
0x000: fe fa 30 02 08 00 08 66 03 00 02 66 03 00 02
```

Before the LAMP instrument can be instructed to include a memory dump packet in the generated TM stream first a time message has to be send to the instrument with the memory dump allowed bit set. The following telecommand stream can be use to initialize the Spacecraft Time to a value of 0x000CF2FF and the Memory dump allowed flag:

```
TIME-MESSAGE::
0x000: fe fa 30 01 3e 00 07 00 0c f2 ff 01 39 00
```

The LAMP instrument will not increment the accepted command counter accepts this message, but the reported time value will now start counting from the value included in this time message.
A memory dump can now be commanded using the memory dump command (note that the instrument has to be in the CHECKOUT state before this command is accepted:

```
LAMP_MEMORY_DUMP::
0x000: fe fa 30 02 14 00 14 66 19 00 05 00 00 00 00 00
0x010: 00 02 00 56 00 00 00 30 19 02 05
```

In response to this telecommand the instrument will generate four memory dump packets that are included in the next two generated telemetry frames. Note again that the accepted command counter will again be incremented. Shown here is an example of the TLM frame including the first generated memory dump packet (with the exact contents of that packet for the current version of the mini LIIS), the two CCSDS primary headers are marked in yellow:

```
0x000: fe fa 30 04 52 01 11 00 00 00 08 81 c0 36 00 73
0x010: 00 00 27 10 30 39 18 c0 30 4c 00 04 00 00 00 02
0x020: 19 ff fe 00 00 00 54 00 00 00 00 01 86 ca 39 48
0x030: 30 39 00 00 7f ff ff ff 00 00 00 02 00 00 02 00
0x040: 00 00 9f aa 1a 9d 16 1d 00 00 00 00 00 00 00 00
0x050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 57
0x060: 57 59 59 74 6c 10 3b ed ff 81 01 00 b8 02 0a 46
0x070: 00 1a 00 5e 78 92 ac c6 e0 fa 14 21 3b a4 f2 60
0x080: 37 dc f1 f2 08 82 c0 00 00 8d 00 0f 42 6c 5b a0
0x090: 00 00 00 00 00 80 56 00 02 2c 27 02 27 76 7f 09
0x0a0: 02 0f 95 02 0e 0e f0 02 2c aa 00 02 3a ea 7f 01
0x0b0: 02 59 95 02 4d 3e 7b 01 7a 82 79 b9 12 5a cd 7b
0x0c0: 01 7a 82 79 bf 12 58 e3 7b 01 7a 82 79 ca 12 58
0x0d0: a2 7b 01 7a 82 79 8b 12 51 d6 90 80 a5 e0 54 1f
0x0e0: fe a3 e0 90 80 a5 12 23 41 90 80 a5 12 22 a8 90
0x0f0: 80 a5 12 23 62 90 80 a5 ee f0 74 81 12 23 ac 90
0x100: 80 a7 12 23 41 90 80 98 12 23 6e 90 80 a7 12 22
0x110: b3 a3 e4 f0 a3 74 73 f0
```

To command the instrument to generate high speed data via the LVDS interface (see appendix B) the instrument has be commanded into acquisition mode. The following three commands can be used to place the instrument into histogram test acquisition mode, with the existing setup the instrument will start generating a histogram test frame once every 5 seconds. The generated high speed test frames are 65536 bytes in size will contain an incrementing number (16-bit) pattern after the first identification word. Note that the confirmation command has to be issued within 30 seconds of the start histogram command:

```
LAMP_DEACTIVATE_HVPS::
0x000: fe fa 30 02 08 00 08 66 0e 00 02 66 0e 00 02
LAMP_START_HISTOGRAM::
0x000: fe fa 30 02 0c 00 0c 66 05 00 03 00 01 00 00 66
0x010: 04 00 03
LAMP_CONFIRM_CRITICAL::
0x000: fe fa 30 02 0c 00 0c 66 04 00 03 66 05 00 00 00
0x010: 01 00 03
```

The first science test frame will be generated after about 30 seconds (as it takes time to generate the test pattern), successive frames will follow at a 5 second rate. The instrument will continue generating high speed science test frames for the next 7000 seconds or until the power of the LIIS is switched off.