XMM Optical Monitor

MULLARD SPACE SCIENCE LABORATORY
UNIVERSITY COLLEGE LONDON

Authors: Philip J. Smith, Matthew C. R. Whillock, Howard E. Huckle, Jeonghee Rho, Rudi Much, Kate G. Adamson

---

XMM-OM USER MANUAL (FM) - PART 2
Experiment Operations Procedures

DOCUMENT: XMM-OM/MSSL/ML/0009.02 interim 1 April 28, 1999

CHANGE RECORD

Contents

List of Tables

List of Figures

1 Introduction and Overview

This is the Experiment Operations Procedures Manual for ^{XMM}OM . It specifies the telecommands to send and telemetry to watch for when performing routine and other ^{XMM}OM procedures.

This document describes the procedures for nominal science and engineering, configuration for orbit phases, maintenance, contingency and commissioning. For each step the database telecommands and params are given, if necessary, the time to wait before the next telecomand, telemetry to expect and for the benefit of ^{XMM}OM consortium software authors, the equivalent VxWorks commands.

The document is intented for use by the XMM MOC and SOC in the writing of their operational spacecraft procedures.

The "Initial Procedures" section describes what should be done during the first turn-on of the Optical Monitor. The "Nominal Procedures" section describes the procedures that should be followed during a normal orbit of switch-on, normal science and switch-off. The "Maintenance Procedures" describes the engineering procedures that will be used early in the mission and if a problem is found. Also included is a section on "Commissioning" to be followed after the initial switch-on and a section on "Error Recovery" describing what to do if the unexpected happens.

2 Reference Documents

- **XMM-OM/MSSL/ML/0010** ^{XMM}OM User Manual Part 3: Telecomand and Telemetry Specification
- **XMM-OM/MSSL/SP/0194** ^{XMM}OM (FM) Integrated Functional Test Procedure
- **XMM-OM/MSSL/ML/0011** ^{XMM}OM User Manual Part 4: ICU-DPU protocol definitions
- **XMM-SOC/TN-0010-SMD** ^{XMM}OM Operational Procedures Handbook
- **OM-MSSL-NCR 94** ^{XMM}OM NCR 94
- **OM-MSSL-NCR 106** ^{XMM}OM NCR 106

Issue	**Date**	**Changes**
0	30-Oct-98	Draft version
1	23-Dec-98	Added wait for Fred. Added ICU mode information at beginning and end of procedures. Added ICU mode change diagram. Corrected window table. Added missing mem window and sci window load tables. Added prime hvon sequence. Added idle before intermediate safe on time-tagged command. Changed timings. Changed the LUT update. Added stop integration commands. Added BPE binning commands. Added flood LED commands to flat-field. Changed frame_time units.
2	26-Mar-99	Blue2 is now the klingon. Change to idle mode at end of "post obs". Added slew. Turn off for perigee. Removed invalid command tests. Added filter wheel hk suspended warnings. Removed redundant powerup from commissioning. Added note on centroid table boundaries. Added substitution heaters on to nominal switch off. Added check HK DPU status word to choose guide stars. Filter to blocked and mode to intermediate safe at end of science data acquisition period. Added PHD. Go to idle after each engineering mode. Removed fixed time_frames from engineering modes. Added error recovery section. Added debug section. Added safing warning. Rewrote initial switch on, science exposure and commissioning sections. Added NCR warnings. Added reset section. Added time verification if Basic checks. Changed filter wheel to datum delay to 25 s. Changed HV timings. Added 100V step. Changed wrong acquisition modes in eng modes. Changed window coords in eng modes. Added move fw to blocked in PPP. Changed the order of things in the PPP. Added Dave at end of eng modes. Removed filter wheel to blocked at end of science data acq. period. Removed cathode to 2 V command. Moved centroid table load to mainenance procedures section.
3	21-Apr-99	Removed Clean Slate's time sync. Corrected code load in initial procedures. Added error recovery detail. Added a turn on tracking at the end of acc and track. Incorrect h510s changed to h150s in hvon procedures.

3 Abbreviations

XMM	X-ray Multi Mirror Mission
OM	Optical Monitor
HK	Housekeeping
KAL	Keep Alive
IFT	Integrated Functional Test
ICU	Instrument Control Unit
DPU	Digital Processing Unit
HV	High Voltage
LUT	Lookup Table
tc	Telecommand
tm	Telemetry
MSSL	Mullard Space Science Lab (University College London)
ESA	European Space Agency
SOC	Science Operations Centre
MCP	Microchannel Plate(s)
FAQ	Field Acquisition
s/c	spacecraft
GS	guide star(s)
RBI	Remote Bus Interface
TMPSU	Telescope module power supply unit
LOS	Loss of Signal

4 Typographical and Other Conventions Used in this Document

4.1 Typographical

- Running text appears in this normal Roman font.
- *Italic font* will indicate a parameter that should not be taken literally.
- A `"typewriter" font` will be used to indicate MSSL EGSE VxWorks commands.
- A **"bold" font** will be used to indicate database entries.

5 Note: Alerts

Please note the following ...

The ground system should always monitor the telemetry stream for exception packets of (type, subtype):

- (3,2): Unsuccessful Command Acceptance
- (3,4): Unsuccessful Command Execution
- (4,2): Exception Report
- (4,3): Major Anomaly Report

as these indicate an error has occured and should only be expected if an intentional bad command is sent as a test.

If the software fails (particularly the ICU software), after taking diagnostics (DMA read of ICU memory addresses fc0$_{hex}$ to fde$_{hex}$ and a21$_{hex}$ to a25$_{hex}$), the ICU should be reset as soon as possible so that it may safe the hardware (see Section 12).

5.1 Safing

In ICU Basic Mode, should the spacecraft issue a safing command, it will not conform to the standard format and the ICU will interpret the packet's CRC as the address to which it will jump leading to unpredictable results. Typically a reset or a watchdog kick (see Appendix C).

6 Initial Procedures

These procedures are intended to be used for the initial switch-on of the Optical Monitor.

6.1 Open Telescope Door

Opening the ^{XMM}OM telescope door is a spacecraft procedure. This should occur at least a week after launch and at least a day after the opening of the last X-ray telescope door.

Must then wait for at least 2 days for outgassing.

6.2 Initial Switch On Procedure (Full Switch On)

6.2.1 Turn on Power

Description	**tc**	**tm**	**params**	**VxWorks**
Spacecraft command: turn on KAL				
Perform the "Nominal Switch On Procedure" (see Section 9.1)				

Note that **the high voltages must not be switched on until at least 2 days after the opening of the door**.

6.2.2 Basic Mode Tests

Description	**tc**	**tm**	**params**	**VxWorks**
Send a valid tc packet	**h1**			`tc_test`
Check HK for good tc packet counter		**90000**	**h5385** should increment by 1	

6.2.3 Time Verification

Description	**tc**	**tm**	**params**	**VxWorks**
Perform Clock Resynchronisation (see Section 10.1)				
Send a tc (10, 5)				`tc_enable_time_verification`
Wait up to 2 s for a tm (10, 5)				
continued on next page				

(continued)

Description	**tc**	**tm**	**params**	**VxWorks**
Check that the contents of the time management report packet are as expected				

6.2.4 Filter Wheel to Datum

Description	**tc**	**tm**	**params**	**VxWorks**
Specify next move of the filter wheel to the datum position	**h7608**			`tc_fw_to_datum`
Command the filter wheel to move	**h5600**			`tc_fw_move`
Wait up to 25 s for a TMPSU event (housekeeping may be suspended for up to 10 s)		**92600**	**h7010="fw @ req pos"**	

6.2.5 Filter Wheel to Blocked

Description	**tc**	**tm**	**params**	**VxWorks**
Specify next move of the filter wheel to the blocked position	**h7604**		**h280="blocked"**	`tc_fw_filter 0`
Command the filter wheel to move	**h5600**			`tc_fw_move`
Wait up to 20 s for a TMPSU event (housekeeping may be suspended for up to 10 s)		**92600**	**h7010="fw @ req pos"**	

6.2.6 Load Code

Description	**tc**	**tm**	**params**	**VxWorks**
Upload ICU operational mode software	multi **h4000** **h4001**		**h500=*address*** **h510=*length***	`<load_icu`

6.2.7 Memory Dump of ICU's Loadable Code

Send dump memory packets to dump out the ICU's operational mode (loadable) code.

Description	**tc**	**tm**	**params**	**VxWorks**
Multiple dump memory requests instruction space	**multi** **h4101**		**h500=*address*** **h510=*length***	

continued on next page

(continued)

Description	tc	tm	params	VxWorks
Multiple dump memory requests operand space	**multi h4100**		**h500=*address* h510=*length***	
Compare the memory dump report tm packets with the memory load packets. Reload full code if a difference is found (see Section 6.2.6).				

6.2.8 Operational Mode

The ICU is in "initial" mode.

Description	tc	tm	params	VxWorks
Mode change to 1 ("full safe")	**h9001**			`tc_mode 1`
Wait up to 20 s				
Check HK		**90000**	**h5405="Operational"**	

The ICU is in "full safe" mode.

6.2.9 Operational Mode Tests

Description	tc	tm	params	VxWorks
Check HK		**90000**	**h5405="Operational"**	
Check HK packet arrives approximately every 10 s		**90000**		
(Thermal control starts automatically)				
Prepare for heater switching: disable standard control algorithm	**h6670**			`tc_heater_stop`
Examine HK		**90000**	**h5240="offoffoffoff"**	
Measure S/C +28V voltage and current. Compare with Table 10				
Start heater contingency task	**h5660**			`tc_heater_config_start`
Switch on heater 1 (Main I/F)	**h7660**		**h380="onoffoffoff"**	`tc_heater_config 8`
Examine HK		**90000**	**h5240="onoffoffoff"**	
Measure S/C +28V voltage and current. Compare with Table 10				
Switch off heater 1	**h7660**		**h380="offoffoffoff"**	`tc_heater_config 0`
Examine HK		**90000**	**h5240="offoffoffoff"**	
Switch on heater 2 (forward tube)	**h7660**		**h380="offonoffoff"**	`tc_heater_config 4`

continued on next page

(continued)

Description	tc	tm	params	VxWorks
Examine HK		**90000**	**h5240="offonoffoff"**	
Measure S/C +28V voltage and current: Compare with Table 10				
Switchoff heater 2	**h7660**		**h380="offoffoffoff"**	`tc_heater_config 0`
Examine HK		**90000**	**h5240="offoffoffoff"**	
Switch on heater 3 (metering rods)	**h7660**		**h380="offoffonoff"**	`tc_heater_config 2`
Examine HK		**90000**	**h5240="offoffonoff"**	
Measure S/C +28V voltage and current: Compare with Table 10				
Switchoff heater 3	**h7660**		**h380="offoffoffoff"**	`tc_heater_config 0`
Examine HK		**90000**	**h5240="offoffoffoff"**	
Switch on heater 4 (secondary mirror)	**h7660**		**h380="offoffoffon"**	`tc_heater_config 1`
Examine HK		**90000**	**h5240="offoffoffon"**	
Measure S/C +28V voltage and current: Compare with Table 10				
Switchoff heater 4	**h7660**		**h380="offoffoffoff"**	`tc_heater_config 0`
Examine HK		**90000**	**h5240="offoffoffoff"**	
Disable heater switching: stop heater contingency task	**h6660**			`tc_heater_config_stop`
Enable standard control algorithm	**h5670**			`tc_heater_start`
Specify next move of the filter wheel to the datum position	**h7608**			`tc_fw_to_datum`
Command the filter wheel to move	**h5600**			`tc_fw_move`
Wait up to 25 s for a TMPSU event (housekeeping may be suspended for up to 15 s)		**92600**	**h7010="fw @ req pos"**	
Examine HK		**90000**	**h5150="seen" h5265=0**	
Command the filter wheel away from datum	**h7606**		**h290=1000**	`tc_fw_rel_steps 1000`
Wait up to 20 s for a TMPSU event (housekeeping may be suspended for up to 10 s)		**92600**	**h7010="fw @ req pos"**	
Examine HK		**90000**	**h5150="seen"** **h5250="not seen"**	
Command the filter wheel to move to the coarse sensor position	**h7609**			`tc_fw_to_coarse`
Command the filter wheel to move	**h5600**			`tc_fw_move`
Wait up to 20 s for a TMPSU event (housekeeping may be suspended for up to 10 s)		**92600**	**h7010="fw @ req pos"**	
Command the filter wheel to move back to the datum position	**h7608**			`tc_fw_to_datum`
Command the filter wheel to move	**h5600**			`tc_fw_move`

continued on next page

(continued)

Description	tc	tm	params	VxWorks
Wait up to 25 s for a TMPSU event (housekeeping may be suspended for up to 15 s)		**92600**	**h7010="fw @ req pos"**	
Examine HK		**90000**	**h5150="seen" h5265=0**	

Table 10: Heater Switching Current Profile

Heater	+28 V Current
1 (Main Interface)	1440mA ± 100 mA
2 (Forward tube)	1740mA ± 100 mA
3 (Metering Rods)	1060mA ± 100 mA
4 (Secondary Mirror)	950mA ± 100 mA
Disable Heater Algorithm	810mA ± 100 mA

6.2.10 DPU

Check the heartbeat count is increasing at 1 every 10 s.

Description	tc	tm	params	VxWorks
Check the heartbeat count is increasing at 1 every 10 s		**h90000**	**h5410**	

6.2.11 Load DPU Code

Description	tc	tm	params	VxWorks
Reset DPU	**h7201**			`tc_reset_dsp`
Wait up to 12 s for alert		**92201**		Expect "Fred from the Dead"
Upload DPU memory load packets	**multi h4020 h4023 h4023 h4027**		**h500=*address* h510=*length***	`<load_dpu`

6.2.12 Memory Dump of DPU's Loadable Code

Send dump memory packets to dump out the DPU's uploadable OS and science code.

Description	tc	tm	params	VxWorks
Multiple Dump Bank 0 requests	**multi h4120**		**h500=*address* h510=*length***	
Multiple Dump Bank 3 requests	**multi h4123**		**h500=*address* h510=*length***	
Multiple Dump Bank 5 requests	**multi h4125**		**h500=*address* h510=*length***	
Multiple Dump Bank 7 requests	**multi h4127**		**h500=*address* h510=*length***	

Compare the memory dump report tm packets with the memory load packets.

6.2.13 Clean Slate

Description	tc	tm	params	VxWorks
Switch to the uploadable code	**h7202**			`tc_load_dpuos`
Wait up to 10 s for DA_DPUOS_READY		**92202**		Expect `''Jim from the Rim''`
Enable red dsp	**h7207**		**h522="Red" h16="Enabled"**	`tc_enbl_dsp red,1`
Wait 10 s				
Enable blue1 dsp	**h7207**		**h522="Blue1" h16="Enabled"**	`tc_enbl_dsp blue1,1`
Wait 10 s				
Enable blue2 dsp	**h7207**		**h522="Blue2" h16="Enabled"**	`tc_enbl_dsp blue2,1`
Wait 10 s				
(Watch out for NCR 94 crash—see Section 11 and Appendix B)				
Select blue2 as the klingon	**h7206**		**h523="blue2"**	`tc_sel_klingon 2`
Initialize DPU	**h7248**			`tc_init_dpu`
Wait up to 25 s for DA_EOT_INIT_DPU event		**92210**		`tm_wait_for_alert DA_EOT_INIT_DPU` ("Dave from the Grave")
Turn on reference frame verbosity (DD_REF)	**h7244**		**h760=3 h16="enable"**	`tc_enbl_verbose 3, 1`

6.2.14 Time Verification

[0]The klingon in DPUOS is the one that 1) finish the memory access and 2) hold on to the bus until the arbitor informs him to give up. The default is white which is the one with the most complicated task. We have to change from the white to blue2 (note not blue 1) so that the BUS IO interrupt will occur always at a know location. This is the result of DPU spontaneous reset at the busy spot, i.e. NCR125.

Description	**tc**	**tm**	**params**	**VxWorks**
Perform Clock Resynchronisation (see Section 10.1)				
Send a tc (10, 5)				`tc_enable_time_verification`
Wait up to 2 s for a tm (10, 5)				
Check that the contents of the time management report packet are as expected				
Check that the DPU time is correct. If not: repeat this time synchronization section				

Begin commissioning (see Section 7).

7 Commissioning

7.1 Tests

7.1.1 Pulse Height Distribution

Description	tc	tm	params	VxWorks
Transition to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`

Perform Engineering Mode 6 exposure (see Section 10.12) with an exposure_time of 30 s and a threshold of 0.

Check the histogram. Check the threshold is nominal.

7.1.2 HV

Note that **the high voltages must not be switched on until at least 2 days after the opening of the Optical Monitor door**.

The following HV ramp-up is for the Prime Channel only.

Description	tc	tm	params	VxWorks
Check HK for temperatures in allowed range and if not, wait		**90000**	**h5135="−8 to +37 °C"**	
Specify V_{mcp23} ramp parameters	**h7140**		**h150="mcp23" h160=100 h175="off"**	`tc_det_load_hv_ramp_params 3, 100, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 20 s for event		**92100**	**h7000="mcp23 ok"**	
Specify V_{mcp23} ramp parameters	**h7140**		**h150="mcp23" h160=1400 h175="off"**	`tc_det_load_hv_ramp_params 3, 1400, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 90 s for event		**92100**	**h7000="mcp23 ok"**	
Specify V_{mcp1} ramp parameters	**h7140**		**h150="mcp1" h160=550 h175="off"**	`tc_det_load_hv_ramp_params 2, 550, 25, 0`
Wait up to 40 s for event		**92100**	**h7000="mcp1 ok"**	
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Specify $V_{cathode}$ ramp parameters	**h7140**		**h150="cathode" h160=396 h175="off"**	`tc_det_load_hv_ramp_params 1, 396, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 25 s for event		**92100**	**h7000="cathode ok"**	
Specify V_{mcp1} ramp parameters	**h7140**		**h150="mcp1" h160=520 h175="off"**	`tc_det_load_hv_ramp_params 2, 520, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`

continued on next page

(continued)

Description	tc	tm	params	VxWorks
Wait up to 20 s for event		**92100**	**h7000="mcp1 ok"**	
Specify V_{mcp23} ramp parameters	**h7140**		**h150="mcp23" h160=1700 h175="off"**	`tc_det_load_hv_ramp_params 3, 1700, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 30 s for event		**92100**	**h7000="mcp23 ok"**	

7.1.3 Pulse Height Distribution with HV

Perform Engineering Mode 6 exposure (see Section 10.12) with an exposure_time of 30 s and a threshold of 0.

Check the histogram. Check the threshold is nominal.

7.1.4 Exposures

Change to "Science" mode:

Description	tc	tm	params	VxWorks
Change to ICU mode 3 ("science")	**h9003**			`tc_mode 3`

Perform "Choose Guide Stars" and "Accumulate and Track" with the filter wheel in the blocked position and the threshold still set to 0. The science window should be 200 x 200 and the exposure_time 100 s.

Examine the image.

Turn on flood LEDs to level 1:

Description	tc	tm	params	VxWorks
Change to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Turn on flood LEDs to level 1	**h7134**		**h1040=1**	`tc_det_flood_led_current 1`
Change to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`
Change to ICU mode 3 ("science")	**h9003**			`tc_mode 3`

Perform "Choose Guide Stars" and "Accumulate and Track" as above.

Examine the image.

Perform an Engineering Mode 6 exposure (see Section 10.12) with an exposure_time of 30 s and a threshold of 0.

Check the histogram and adjust threshold if necessary.

Perform an Engineering Mode 3 exposure (see Section 10.9) with an exposure_time of 300 s and a threshold of 15.

Check boundary values are nominal.

Look Kate . . . 4 ks

Perform an Engineering Mode 5 (centroiding check) exposure (see Section 10.11) with an exposure_time of 4000 s and a threshold of 15.

Point XMM at the OM boresight calibration field.

Perform "Acquisition", "Choose Guide Stars" and "Accumulate and Track" with the V filter, a science window of 200 x 200 and an exposure_time of 100 s.

Examine image, guide stars etc.

Perform an Engineering Mode 4 exposure (see Section 10.10) with V filter and an exposure_time of 1000 s.

(Natural break point, if necessary.)

7.2 Focus

Set up with appropriate science windows on a test star and use the V filter. Perform "Acquisition", "Choose Guide Stars" and "Accumulate and Track".

Check the image. If necessary the following focusing tests may need to be performed.

Do the same with the magnifier instead of the V filter.

7.2.1 Focus with V Filter

If the focus is too bad, perform "Choose Guide Stars" and "Accumulate and Track" with the filter wheel in the V position and the focusing heaters at various settings. The science windows should be suitable for the field. This process could take many hours. It may take an hour between heater settings for the temperature to stabilize.

Anaylze images and adjust heaters for best focus.

7.2.2 Focus with Magnifier

Repeat the focus process above but with the magnifier instead of the V filter. This could take many hours.

Anaylze images and adjust heaters for best focus.

Perform engineering mode 4 with an exposure time of 1000 s.

7.3 Boresight

The above fullframe image should be checked to assess the boresight alignment.

7.4 Filters

Perform a standard exposure with each filter. Check the images.

7.5 Flatfield

Perform a standard flatfield (see Section 9.11).

7.6 End of Tests

Description	tc	tm	params	VxWorks
Perfom nominal switch off (see Section 9.2)				

The ICU is off.

8 Calibration and PV

See Rudi.

9 Nominal Procedures

These procedures are intended to be used in normal use during an XMM orbit.

9.1 Nominal Switch On

Description	tc	tm	params	VxWorks
Spacecraft procedure: Switch off substitution heater				
Spacecraft procedure: Switch on 28V.				
Wait up to 20 s for a bootstrap report		**92703**	"bootstrap report event packet"	
Check the HK: expect a HK packet every 10 s		**90000**		
In the HK packets, look for mode and heater status		**90000**	**h5395="Initial"** **h5240="OnOnOffOff"** **h5255="Disabled"**	
Upload "initial" mode patches using RBI write spacecraft commands				

The ICU will be in "initial" mode.

9.2 Nominal Switch Off

N.B. High voltages must be ramped down slowly and safely before switching off main power, as described in this procedure.

The ICU will be in “idle”, “science”, “engineering” or “intermediate safe” mode.

Description	**tc**	**tm**	**params**	**VxWorks**
Specify $V_{cathode}$ ramp parameters	**h7140**		**h510=“cathode”, h160=0, h175=“off”**	`tc_det_load_hv_ramp_params 1,0,25,0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 120 s for event		**92100**	**h7000=“cathode ok”**	
Specify V_{mcp1} ramp parameters	**h7140**		**h510=“mcp1”, h160=0, h175=“off”**	`tc_det_load_hv_ramp_params 2,0,25,0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 120 s for event		**92100**	**h7000=“mcp1 ok”**	
Specify V_{mcp23} ramp parameters	**h7140**		**h510=“mcp23, h160=0, h175=“off”**	`tc_det_load_hv_ramp_params 3,0,25,0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 120 s for event		**92100**	**h7000=“mcp23 ok”**	
Spacecraft command: Send an RBI suspend	**D7200**			`suspend`
Spacecraft procedure: switch off +28 V				
Spacecraft procedure: switch on substitution heaters				

9.3 Transition to Safe Mode

Before this procedure, the ICU will be in "initial", "idle", "intermediate safe", "science", or "engineering" mode.

This is a initiated by a **tc h9001** either autonomously from the spacecraft or as a telecommand from the ground.

General case:

Description	tc	tm	params	VxWorks
Mode change to 1 ("full safe")	**h9001**			`tc_mode 1`
Wait up to 160 s				
Check HK		**90000**	**h5395="Safe"**	

When in "Initial mode":

Description	tc	tm	params	VxWorks
Mode change to 1 ("full safe")	**h9001**			`tc_mode 1`
Wait up to 2 s				
Check HK		**90000**	**h5395="Safe"**	

When in "Intermediate Safe" mode with the high voltages on:

Description	tc	tm	params	VxWorks
Mode change to 1 ("full safe")	**h9001**			`tc_mode 1`
Wait up to 20 s for mode change event packet		**92100**	**h7000="MCP1 ok"**	
Wait up to 100 s for mode change event packet		**92100**	**h7000="MCP2,3 ok"**	
Wait up to 40 s for 0, 1 or 2 filter wheel event packets		**92600**	**h7010="fw at req pos"**	
Check HK		**90000**	**h5395="Safe"**	

When in "Intermediate Safe" mode with the high voltages off:

Description	tc	tm	params	VxWorks
Mode change to 1 ("full safe")	**h9001**			`tc_mode 1`
Wait up to 40 s for 0, 1 or 2 filter wheel event packets		**92600**	**h7010="fw at req pos"**	
Check HK		**90000**	**h5395="Safe"**	

When in "idle", "science" or "engineering" modes with the high voltages on:

Description	tc	tm	params	VxWorks
Mode change to 1 ("full safe")	**h9001**			`tc_mode 1`
Wait up to 20 s for mode change event packet		**92100**	**h7000="cathode ok"**	
Wait up to 20 s for mode change event packet		**92100**	**h7000="MCP1 ok"**	
Wait up to 100 s for mode change event packet		**92100**	**h7000="MCP2,3 ok"**	
Wait up to 40 s for 0, 1 or 2 filter wheel event packets		**92600**	**h7010="fw at req pos"**	
Check HK		**90000**	**h5395="Safe"**	

When in "idle", "science" or "engineering" modes with the high voltages off:

Description	tc	tm	params	VxWorks
Mode change to 1 ("full safe")	**h9001**			`tc_mode 1`
Wait up to 20 s for filter wheel event packet		**92600**	**h7010="fw at req pos"**	
If filter wheel has not yet been to datum, wait up to 20 s for another filter wheel event packet		**92600**	**h7010="fw at req pos"**	
Check HK		**90000**	**h5395="Safe"**	

9.4 Example Observation

The example observation shown in Table 29 consists of an acquisition phase followed by 2 exposures. In turn, each exposure consists of a choose guide star phase followed by an accumulation and tracking phase. Science data is not available for downlinking until the end of the first exposure. The status of science data, filter wheel and detector during the course of the observation is summarized Table 29. This example can be generalized to a n-exposure sequence by repeating the "between exposures" and "Exposure 2" (with suitable changes in the uplinked configuration) phases as often as required.

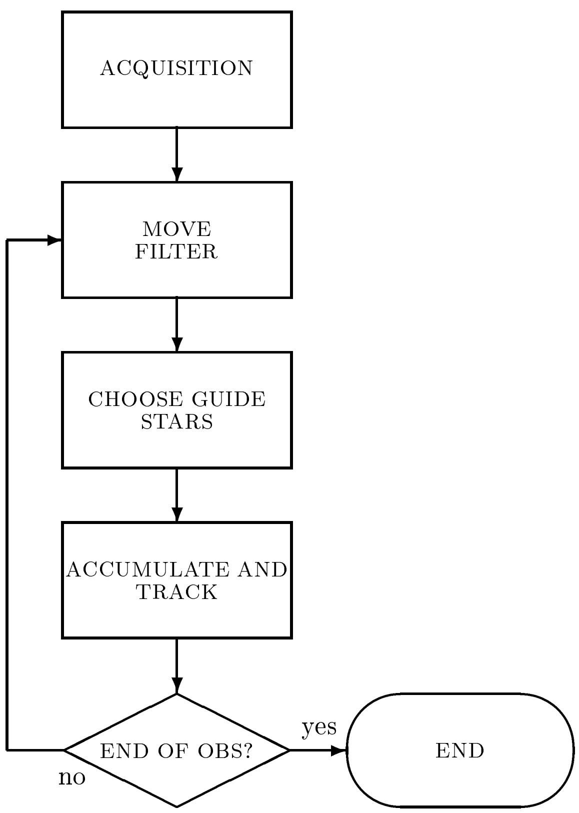
Table 29: Example Observation

Observation Phase	**Associated Command Sequences**	**Detector Status**		**Filter Wheel Position**	**Science Data Status**	
		Windows	**Resolution**		**Priority**	**Regular**
Acquisition	Acquisition	Four abutting windows covering the whole detector	Low	V Filter	–	–
Exposure 1	Choose Guide Stars			Observer Requested	Exposure 1 Downloading	–
	Accumulate and Track	Observer requested + guiding	High			
Between Exposures	–	HV on Data not sent to DPU		Last Observer Request	–	
Exposure2	Choose Guide Stars	Four abutting windows covering the whole detector	Low	Observer Requested	Exposure2 Downloading	Exposure 1 Downloading
	Accumulate and Track	Observer requested + guiding	High			
Post Observation	Post Observation	HV On Data not sent to DPU		Blocked	–	Exposure2 Downloading

9.5 Overall Flow Diagram

The overall flow diagram is shown in Figure 1.

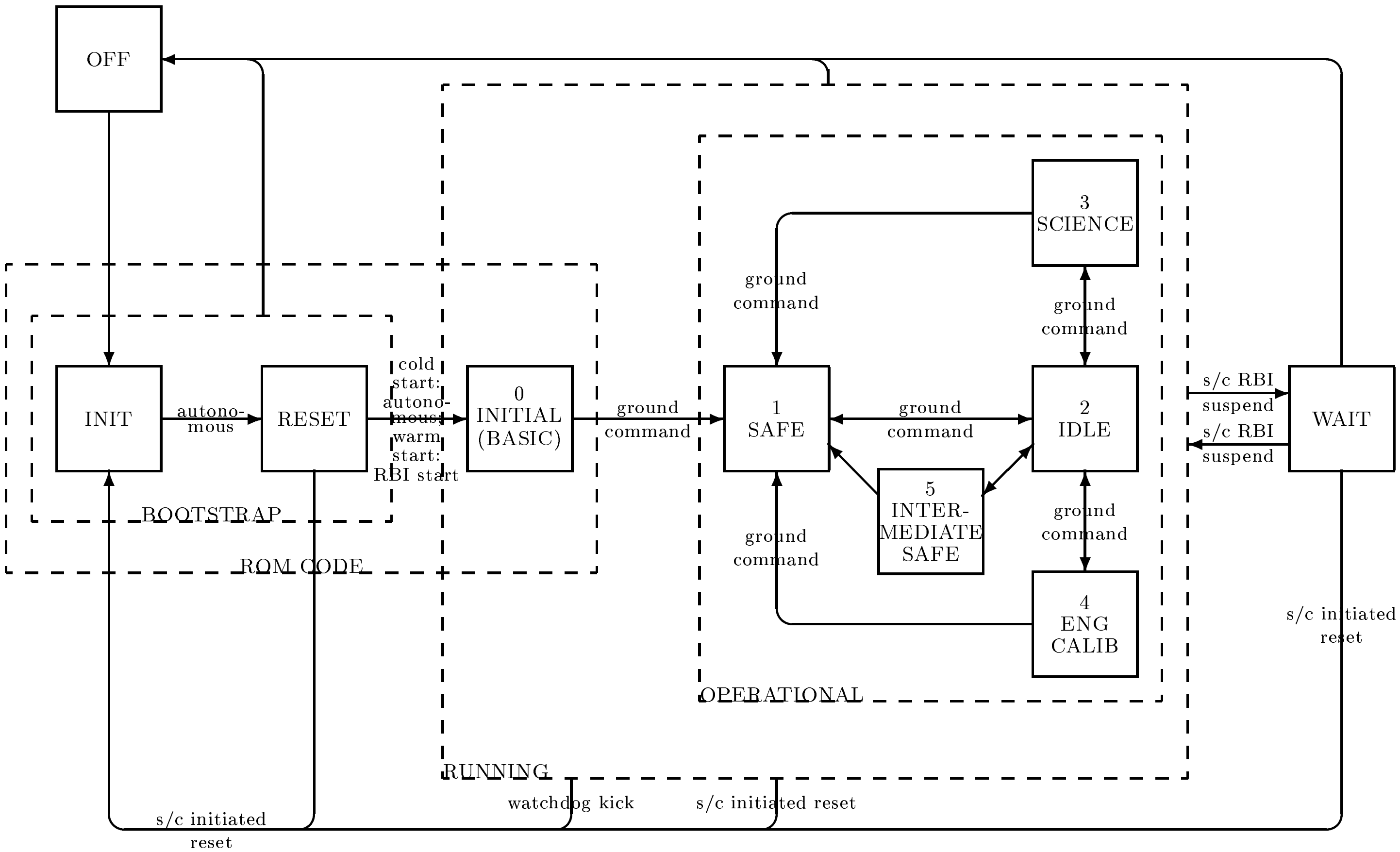
Figure 1: Overall Flow Diagram



9.6 OM FM Mode Transitions

The OM FM mode transitions are shown in Figure 2.

Figure 2: OM FM Mode Transitions



9.7 Acquisition

The ICU is in “science” mode.

Description	**tc**	**tm**	**params**	**VxWorks**
Set acquisition mode	**h7130**		**“low res win”**	`tc_det_acq_mode 0`
Load window table parameters	**h7110**		***Table 31***	`det_reset_wind ;` `det_wind` *arg, arg, arg, arg ...*
Start load window table task	**h5110**			`tc_det_load_windows 4`
Wait up to 30 s for detector event		**92100**	**“win tab load”**	
Set filter wheel number for the V filter	**h7604**		**h280=“V”**	`tc_fw_filter 1`
Move the filter wheel	**h5600**			`tc_fw_move`
Wait up to 20 s for TMPSU event (housekeeping may be suspended for up to 10 s)		**92600**	**h7010=“FW @ req pos”**	
Set BPE binning 0	**h7257**		**h540=“No binning”**	`tc_set_bpe 0`
Set the exposure id	**h7238**		**h530=*exposure_number***	`tc_set_exp_no` *exposure_id*
Ensure frame tags are enabled	**h7135**		**h16=“enabled”**	`tc_det_enbl_frame_tag 1`
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *threshold*
Load the reference stars	**h7240**		***Table 32***	`tc_prog_ref_stars` *file*
Set the DPU frame time (in units of 1/1024 s)	**h7236**		**h710=*frame_time***	`tc_set_frame_time` *frame_time*
Prepare the DPU for acquiring low resolution data	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
Start the detector integration	**h5130**			`tc_det_start_int`
Initiate acquire field	**h7250**			`tc_send_command IC_ACQUIRE_FLD` or `tc_acquire_field`
Wait for (*frame_time* + 1) s for the end of integration		**92211**		
Stop integration		**h6130**		`tc_det_stop_int`
Wait 60 s for DA_EOT_ACQUIRE_FLD		**92211**		`tm_wait_for_alert DA_EOT_ACQUIRE_FLD`
Receive priority science data (acquisition data)		**97402**		

The ICU is in “science” mode.

Table 31: Window Table 1024 x 1024

Description	Params	Value	
		Prime	Redundant
Verification	h10	"enable"	"enable"
Number of windows	h40	4	4
Window 1 Xlow	h50	52	64
Window 1 Ylow	h51	27	15
Window 1 Xsize	h52	128	128
Window 1 Ysize	h53	128	128
Window 2 Xlow	h50	52	64
Window 2 Ylow	h51	155	143
Window 2 Xsize	h52	128	128
Window 2 Ysize	h53	128	128
Window 3 Xlow	h50	180	192
Window 3 Ylow	h51	27	15
Window 3 Xsize	h52	128	128
Window 3 Ysize	h53	128	128
Window 4 Xlow	h50	180	192
Window 4 Ylow	h51	155	143
Window 4 Xsize	h52	128	128
Window 4 Ysize	h53	128	128

Table 32: Reference Stars

Description	PARAMS
Verification	**h10**
Number of reference stars	**h1160**
X co-ordinate of reference star 1	**h1095**
Y co-ordinate of reference star 1	**h1096**
X co-ordinate of reference star 2	**h1097**
Y co-ordinate of reference star 2	**h1098**
X co-ordinate of reference star 3	**h1099**
Y co-ordinate of reference star 3	**h1100**
X co-ordinate of reference star 4	**h1101**
Y co-ordinate of reference star 4	**h1102**
X co-ordinate of reference star 5	**h1103**
Y co-ordinate of reference star 5	**h1104**
X co-ordinate of reference star 6	**h1105**
Y co-ordinate of reference star 6	**h1106**
X co-ordinate of reference star 7	**h1107**
Y co-ordinate of reference star 7	**h1108**
X co-ordinate of reference star 8	**h1109**
Y co-ordinate of reference star 8	**h1110**
X co-ordinate of reference star 9	**h1111**
Y co-ordinate of reference star 9	**h1112**
X co-ordinate of reference star 10	**h1113**
Y co-ordinate of reference star 10	**h1114**
X co-ordinate of reference star 11	**h1115**
Y co-ordinate of reference star 11	**h1116**
X co-ordinate of reference star 12	**h1117**
Y co-ordinate of reference star 12	**h1118**
X co-ordinate of reference star 13	**h1119**
Y co-ordinate of reference star 13	**h1120**
X co-ordinate of reference star 14	**h1121**
Y co-ordinate of reference star 14	**h1122**
X co-ordinate of reference star 15	**h1123**
Y co-ordinate of reference star 15	**h1124**
X co-ordinate of reference star 16	**h1125**
Y co-ordinate of reference star 16	**h1126**
X co-ordinate of reference star 17	**h1127**
Y co-ordinate of reference star 17	**h1128**
X co-ordinate of reference star 18	**h1129**
Y co-ordinate of reference star 18	**h1130**
X co-ordinate of reference star 19	**h1131**
Y co-ordinate of reference star 19	**h1132**
X co-ordinate of reference star 20	**h1133**

coninued on next page

(continued)

Description	PARAMS
Y co-ordinate of reference star 20	**h1134**
X co-ordinate of reference star 21	**h1135**
Y co-ordinate of reference star 21	**h1136**
X co-ordinate of reference star 22	**h1137**
Y co-ordinate of reference star 22	**h1138**
X co-ordinate of reference star 23	**h1139**
Y co-ordinate of reference star 23	**h1140**
X co-ordinate of reference star 24	**h1141**
Y co-ordinate of reference star 24	**h1142**
X co-ordinate of reference star 25	**h1143**
Y co-ordinate of reference star 25	**h1144**
X co-ordinate of reference star 26	**h1145**
Y co-ordinate of reference star 26	**h1146**
X co-ordinate of reference star 27	**h1147**
Y co-ordinate of reference star 27	**h1148**
X co-ordinate of reference star 28	**h1149**
Y co-ordinate of reference star 28	**h1150**
X co-ordinate of reference star 29	**h1151**
Y co-ordinate of reference star 29	**h1152**
X co-ordinate of reference star 30	**h1153**
Y co-ordinate of reference star 30	**h1154**
X co-ordinate of reference star 31	**h1155**
Y co-ordinate of reference star 31	**h1156**
X co-ordinate of reference star 32	**h1157**
Y co-ordinate of reference star 32	**h1158**

9.8 Move Filter

The ICU is in "science" mode.

Description	tc	tm	params	VxWorks
Set filter number	**h7604**		**h280=*filter***	`tc_fw_filter` *`filter`*
Move the filter wheel	**h5600**			`tc_fw_move`
Wait up to 20 s for TMPSU event (housekeeping may be suspended for up to 10 s)		**92600**	**h7010="FW @ req pos"**	

The ICU is in "science" mode.

9.9 Choose Guide Stars

The ICU is in "science" mode.

Description	tc	tm	params	VxWorks
Set acquisition mode	**h7130**		**"low res win"**	`tc_det_acq_mode 0`
Load window table parameters	**h7110**		***Table 31***	`det_reset_wind ;` `det_wind` *arg, arg, arg, arg ...*
Set the exposure id	**h7238**		**h530=*exposure_number***	`tc_set_exp_no` *exposure_id*
Send IC_INIT_EXP	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
Ensure frame tags are enabled	**h7135**		**h16="enabled"**	`tc_det_enbl_frame_tag 1`
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *threshold*
Set the DPU frame time (in units of 1/1024 s)	**h7236**		**h710=*frame_time***	`tc_set_frame_time` *frame_time*
Load the memory windows	**h7242**		***Table 35***	`tc_prog_mem_wind` ''*file*''
Load the science windows	**h7243**		***Table 36***	`tc_prog_sci_wind` ''*file*''
Set the exposure duration (in frames)	**h7237**		**h720=*number_of_frames***	`tc_set_exposure_time`
Set BPE binning 0	**h7257**		**h540="No binning"**	`tc_set_bpe 0`
Start the detector integration task	**h5130**			`tc_det_start_int`
Enable choose guide stars task	**h7251**			`tc_send_command IC_CHOOSE_GS`
Wait up to (*frame_time* + 40 + 40) s for DPU event report DA_EOT_CHOOSE_GS		**92212**		`tm_wait_for_alert DA_EOT_CHOOSE_GS`
Look for priority science data (window data)		**97401**		
Look for window table loaded event		**92100**	**h7000="win tab load"**	
Stop integration		**h6130**		`tc_det_stop_int`

The ICU is in "science" mode.

Table 36: Load Science Windows

Description	Params
Number of science windows (max 5)	**h1161**
Window 1 bottom left x coord	**h1050**
Window 1 bottom left y coord	**h1051**
Window 1 x size (centroided pixels)	**h1052**
Window 1 y size (centroided pixels)	**h1053**
coninued on next page	

(continued)

Description	PARAMS
Window 1 operation mode	**h1070**
Window 1 mode-dependent parameter	**h1071**
Window 1 mode-dependent parameter	**h1072**
Window 1 mode-dependent parameter	**h1073**
Window 1 mode-dependent parameter	**h1074**
Window 1 address in DPU RAM	
Window 1 memory window ID	
Window 2 bottom left x coord	**h1054**
Window 2 bottom left y coord	**h1055**
Window 2 x size (centroided pixels)	**h1056**
Window 2 y size (centroided pixels)	**h1057**
Window 2 operation mode	**h1075**
Window 2 mode-dependent parameter	**h1076**
Window 2 mode-dependent parameter	**h1077**
Window 2 mode-dependent parameter	**h1078**
Window 2 mode-dependent parameter	**h1079**
Window 2 address in DPU RAM	
Window 2 memory window ID	
Window 3 bottom left x coord	**h1058**
Window 3 bottom left y coord	**h1059**
Window 3 x size (centroided pixels)	**h1060**
Window 3 y size (centroided pixels)	**h1061**
Window 3 operation mode	**h1080**
Window 3 mode-dependent parameter	**h1081**
Window 3 mode-dependent parameter	**h1082**
Window 3 mode-dependent parameter	**h1083**
Window 3 mode-dependent parameter	**h1084**
Window 3 address in DPU RAM	
Window 3 memory window ID	
Window 4 bottom left x coord	**h1062**
Window 4 bottom left y coord	**h1063**
Window 4 x size (centroided pixels)	**h1064**
Window 4 y size (centroided pixels)	**h1065**
Window 4 operation mode	**h1085**
Window 4 mode-dependent parameter	**h1086**
Window 4 mode-dependent parameter	**h1087**
Window 4 mode-dependent parameter	**h1088**
Window 4 mode-dependent parameter	**h1089**
Window 4 address in DPU RAM	
Window 4 memory window ID	
Window 5 bottom left x coord	**h1066**

coninued on next page

(continued)

Description	**PARAMS**
Window 5 bottom left y coord	**h1067**
Window 5 x size (centroided pixels)	**h1068**
Window 5 y size (centroided pixels)	**h1069**
Window 5 operation mode	**h1090**
Window 5 mode-dependent parameter	**h1091**
Window 5 mode-dependent parameter	**h1092**
Window 5 mode-dependent parameter	**h1093**
Window 5 mode-dependent parameter	**h1094**
Window 5 address in DPU RAM	
Window 5 memory window ID	

Table 35: Load Memory Windows

Description	Params
Number of memory windows (max 5)	**h1159**
Window 1 bottom left x coord	**h1050**
Window 1 bottom left y coord	**h1051**
Window 1 x size (centroided pixels)	**h1052**
Window 1 y size (centroided pixels)	**h1053**
Window 1 address in DPU RAM	
Window 2 bottom left x coord	**h1054**
Window 2 bottom left y coord	**h1055**
Window 2 x size (centroided pixels)	**h1056**
Window 2 y size (centroided pixels)	**h1057**
Window 2 address in DPU RAM	
Window 3 bottom left x coord	**h1058**
Window 3 bottom left y coord	**h1059**
Window 3 x size (centroided pixels)	**h1060**
Window 3 y size (centroided pixels)	**h1061**
Window 3 address in DPU RAM	
Window 4 bottom left x coord	**h1062**
Window 4 bottom left y coord	**h1063**
Window 4 x size (centroided pixels)	**h1064**
Window 4 y size (centroided pixels)	**h1065**
Window 4 address in DPU RAM	
Window 5 bottom left x coord	**h1066**
Window 5 bottom left y coord	**h1067**
Window 5 x size (centroided pixels)	**h1068**
Window 5 y size (centroided pixels)	**h1069**
Window 5 address in DPU RAM	

9.10 Accumulate and Track

The ICU is in "science" mode.

Description	tc	tm	params	VxWorks
Set the acquisition mode to high resolution windowed	**h7130**		**h110="Hi Res Win"**	`tc_det_acq_mode 2`
Set BPE binning 0	**h7257**		**h540="No binning"**	`tc_set_bpe 0`
Enable frame tags	**h7135**		**h16="enabled"**	`tc_det_enbl_frame_tag 1`
Set threshold	**h7131**		**h120=30**	`tc_det_event_threshold 15`
Start the detector integration task	**h5130**			`tc_det_start_int`
Enable track guide stars task	**h7252**			`tc_send_command IC_TRACK_GS`
Wait up to (frame_time × (exposure_time + 1) + 10) s for DPU event report end of exposure		**92205**		`tm_wait_for_alert DA_COMPLETE_EXP`
Turn on tracking	**h7256**		**h1040="on"**	`tc_set_trk 1`

The ICU is in "science" mode.

9.11 Flatfielding

The ICU is in "science" mode. The filter wheel must be in the "blocked" position.

Description	tc	tm	params	VxWorks
Set the filter wheel to blocked	**h7604**		**h280="Blocked"**	`tc_fw_filter 0`
Move the filter wheel (housekeeping may be suspended for up to 10 s)	**h5600**			`tc_fw_move`
Wait up to 20 s for TMPSU event report		**92600**	**h7010="fw @ req pos"**	
Check HK		**90000**	**h5265=1200 ("blocked")**	
Turn off tracking	**h7256**		**h1040="off"**	`tc_set_trk 0`
Set BPE binning	**h7257**		**h540="binning"**	`tc_set_bpe 1`
Change to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Turn on flood LEDs to level 4 (approx 20 000 count s^{-1})	**h7134**		**h1040=*current***	`tc_det_flood_led_current` *current*
Change to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`
Change to ICU mode 3 ("science")	**h9003**			`tc_mode 3`

Run the following standard exposure ("Choose Guide Stars followed by "Accumulate and Track") 4 times, the first time with the science windows in Table 42, the second time with the science windows in Table 43, the third time with the science windows in Table 44, and the fourth and last time with the science windows in Table 45.

Description	tc	tm	params	VxWorks
Set the acquisition mode to high resolution windowed	**h7130**		**h110="Hi Res Win"**	`tc_det_acq_mode 2`
Load window table parameters	**h7110**		***Table 31***	`det_reset_wind ;` `det_wind` *arg, arg, arg, arg ...*
Set exposure id	**h7238**		**h530=*exposure_id***	`tc_set_exp_no` *exposure_id*
Send IC_INIT_EXP	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
Ensure frame tags are enabled	**h7135**		**h16="enabled"**	`tc_det_enbl_frame_tag 1`
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *threshold*
Set the DPU frame time (in units of 1/1024 s) to 25	**h7236**		**h710=25**	`tc_set_frame_time 25`
Load the memory windows	**h7242**		***Table 41***	`tc_prog_mem_wind` *''file''*
Load the science windows	**h7243**		***Tables 42 or 43 or 44 or 45***	`tc_prog_sci_wind` *''file''*

continued on next page

(continued)

Description	tc	tm	params	VxWorks
Set exposure time, in units of frames to 20	**h7237**		**h720=20**	`tc_set_exposure_time 20`
Set BPE binning	**h7257**		**h540="binning"**	`tc_set_bpe 1`
Start sending events from mic	**h5130**			`tc_det_start_int`
Invoking the choose_guide_stars task on the DPU	**h7251**			`tc_send_command IC_CHOOSE_GS`
Wait up to $(frame\_time + 40 + 40)$ s for alert DA_EOT_CHOOSE_GS		**92212**		`tm_wait_for_alert DA_EOT_CHOOSE_GS`
Stop integration		**h6130**		`tc_det_stop_int`
Set acquisition mode	**h7130**		**h110="hi res win"**	`tc_det_acq_mode 2`
Set BPE binning 1	**h7257**		**h540="binning"**	`tc_set_bpe 1`
Ensure frame tags are enabled	**h7135**		**h16="enabled"**	`tc_det_enbl_frame_tag 1`
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *threshold*
Start sending events from mic	**h5130**			`tc_det_start_int`
Start the exposure	**h7252**			`tc_send_command IC_TRACK_GS`
Wait for up to $(frame\_time \times (exposure\_time + 1)) + 10)$ s for alert DA_COMPLETE_EXP		**92209**		`tm_wait_for_alert DA_COMPLETE_EXP`

Description	tc	tm	params	VxWorks
Turn on tracking	**h7256**		**h1040="on"**	`tc_set_trk 1`
Set BPE binning 0	**h7257**		**h540="No binning"**	`tc_set_bpe 0`
Change to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Turn off flood LEDs	**h7134**		**h1040=0**	`tc_det_flood_led_current 0`
Change to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`
Change to ICU mode 3 ("science")	**h9003**			`tc_mode 3`

The ICU is in "science" mode.

Table 42: Flatfield Science Windows 1

Description	Params
Number of science windows	**h1161 = 4**
Window 1 bottom left x coord	**h1050 = 0**
Window 1 bottom left y coord	**h1051 = 0**

coninued on next page

(continued)

Description	**Params**
Window 1 x size (centroided pixels)	**h1052 = 64**
Window 1 y size (centroided pixels)	**h1053 = 1024**
Window 1 operation mode	**h1070 = 0**
Window 1 mode-dependent parameter	**h1071 = 0**
Window 1 mode-dependent parameter	**h1072 = 0**
Window 1 mode-dependent parameter	**h1073 = 0**
Window 1 mode-dependent parameter	**h1074 = 0**
Window 1 address in DPU RAM	0
Window 1 memory window ID	0
Window 2 bottom left x coord	**h1054 = 64**
Window 2 bottom left y coord	**h1055 = 0**
Window 2 x size (centroided pixels)	h**1056 = 64**
Window 2 y size (centroided pixels)	**h1057 = 1024**
Window 2 operation mode	**h1075 = 0**
Window 2 mode-dependent parameter	**h1076 = 0**
Window 2 mode-dependent parameter	**h1077 = 0**
Window 2 mode-dependent parameter	**h1078 = 0**
Window 2 mode-dependent parameter	**h1079 = 0**
Window 2 address in DPU RAM	0
Window 2 memory window ID	0
Window 3 bottom left x coord	**h1058 = 128**
Window 3 bottom left y coord	**h1059 = 0**
Window 3 x size (centroided pixels)	**h1060 = 64**
Window 3 y size (centroided pixels)	**h1061 = 1024**
Window 3 operation mode	**h1080 = 0**
Window 3 mode-dependent parameter	**h1081 = 0**
Window 3 mode-dependent parameter	**h1082 = 0**
Window 3 mode-dependent parameter	**h1083 = 0**
Window 3 mode-dependent parameter	**h1084 = 0**
Window 3 address in DPU RAM	0
Window 3 memory window ID	0
Window 4 bottom left x coord	**h1062 = 192**
Window 4 bottom left y coord	**h1063 = 0**
Window 4 x size (centroided pixels)	**h1064 = 64**
Window 4 y size (centroided pixels)	**h1065 = 1024**
Window 4 operation mode	**h1085 = 0**
Window 4 mode-dependent parameter	**h1086 = 0**
Window 4 mode-dependent parameter	**h1087 = 0**
Window 4 mode-dependent parameter	**h1088 = 0**
Window 4 mode-dependent parameter	**h1089 = 0**
Window 4 address in DPU RAM	0

coninued on next page

(continued)

Description	Params
Window 4 memory window ID	0

Table 43: Flatfield Science Windows 2

Description	Params
Number of science windows	**h1161 = 4**
Window 1 bottom left x coord	**h1050 = 256**
Window 1 bottom left y coord	**h1051 = 0**
Window 1 x size (centroided pixels)	**h1052 = 64**
Window 1 y size (centroided pixels)	**h1053 = 1024**
Window 1 operation mode	**h1070 = 0**
Window 1 mode-dependent parameter	**h1071 = 0**
Window 1 mode-dependent parameter	**h1072 = 0**
Window 1 mode-dependent parameter	**h1073 = 0**
Window 1 mode-dependent parameter	**h1074 = 0**
Window 1 address in DPU RAM	0
Window 1 memory window ID	0
Window 2 bottom left x coord	**h1054 = 320**
Window 2 bottom left y coord	**h1055 = 0**
Window 2 x size (centroided pixels)	**h1056 = 64**
Window 2 y size (centroided pixels)	**h1057 = 1024**
Window 2 operation mode	**h1075 = 0**
Window 2 mode-dependent parameter	**h1076 = 0**
Window 2 mode-dependent parameter	**h1077 = 0**
Window 2 mode-dependent parameter	**h1078 = 0**
Window 2 mode-dependent parameter	**h1079 = 0**
Window 2 address in DPU RAM	0
Window 2 memory window ID	0
Window 3 bottom left x coord	**h1058 = 384**
Window 3 bottom left y coord	**h1059 = 0**
Window 3 x size (centroided pixels)	**h1060 = 64**
Window 3 y size (centroided pixels)	**h1061 = 1024**
Window 3 operation mode	**h1080 = 0**
Window 3 mode-dependent parameter	**h1081 = 0**
Window 3 mode-dependent parameter	**h1082 = 0**
Window 3 mode-dependent parameter	**h1083 = 0**
Window 3 mode-dependent parameter	**h1084 = 0**
Window 3 address in DPU RAM	0
Window 3 memory window ID	0
Window 4 bottom left x coord	**h1062 = 448**
Window 4 bottom left y coord	**h1063 = 0**

coninued on next page

(continued)

Description	Params
Window 4 x size (centroided pixels)	**h1064 = 64**
Window 4 y size (centroided pixels)	**h1065 = 1024**
Window 4 operation mode	**h1085 = 0**
Window 4 mode-dependent parameter	**h1086 = 0**
Window 4 mode-dependent parameter	**h1087 = 0**
Window 4 mode-dependent parameter	**h1088 = 0**
Window 4 mode-dependent parameter	**h1089 = 0**
Window 4 address in DPU RAM	0
Window 4 memory window ID	0

Table 44: Flatfield Science Windows 3

Description	Params
Number of science windows	**h1161 = 4**
Window 1 bottom left x coord	**h1050 = 512**
Window 1 bottom left y coord	**h1051 = 0**
Window 1 x size (centroided pixels)	**h1052 = 64**
Window 1 y size (centroided pixels)	**h1053 = 1024**
Window 1 operation mode	**h1070 = 0**
Window 1 mode-dependent parameter	**h1071 = 0**
Window 1 mode-dependent parameter	**h1072 = 0**
Window 1 mode-dependent parameter	**h1073 = 0**
Window 1 mode-dependent parameter	**h1074 = 0**
Window 1 address in DPU RAM	0
Window 1 memory window ID	0
Window 2 bottom left x coord	**h1054 = 576**
Window 2 bottom left y coord	**h1055 = 0**
Window 2 x size (centroided pixels)	**h1056 = 64**
Window 2 y size (centroided pixels)	**h1057 = 1024**
Window 2 operation mode	**h1075 = 0**
Window 2 mode-dependent parameter	**h1076 = 0**
Window 2 mode-dependent parameter	**h1077 = 0**
Window 2 mode-dependent parameter	**h1078 = 0**
Window 2 mode-dependent parameter	**h1079 = 0**
Window 2 address in DPU RAM	0
Window 2 memory window ID	0
Window 3 bottom left x coord	**h1058 = 640**
Window 3 bottom left y coord	**h1059 = 0**
Window 3 x size (centroided pixels)	**h1060 = 64**
Window 3 y size (centroided pixels)	**h1061 = 1024**
Window 3 operation mode	**h1080 = 0**

coninued on next page

(continued)

Description	Params
Window 3 mode-dependent parameter	**h1081 = 0**
Window 3 mode-dependent parameter	**h1082 = 0**
Window 3 mode-dependent parameter	**h1083 = 0**
Window 3 mode-dependent parameter	**h1084 = 0**
Window 3 address in DPU RAM	0
Window 3 memory window ID	0
Window 4 bottom left x coord	**h1062 = 704**
Window 4 bottom left y coord	**h1063 = 0**
Window 4 x size (centroided pixels)	**h1064 = 64**
Window 4 y size (centroided pixels)	**h1065 = 1024**
Window 4 operation mode	**h1085 = 0**
Window 4 mode-dependent parameter	**h1086 = 0**
Window 4 mode-dependent parameter	**h1087 = 0**
Window 4 mode-dependent parameter	**h1088 = 0**
Window 4 mode-dependent parameter	**h1089 = 0**
Window 4 address in DPU RAM	0
Window 4 memory window ID	0

Table 45: Flatfield Science Windows 4

Description	Params
Number of science windows	**h1161 = 4**
Window 1 bottom left x coord	**h1050 = 768**
Window 1 bottom left y coord	**h1051 = 0**
Window 1 x size (centroided pixels)	**h1052 = 64**
Window 1 y size (centroided pixels)	**h1053 = 1024**
Window 1 operation mode	**h1070 = 0**
Window 1 mode-dependent parameter	**h1071 = 0**
Window 1 mode-dependent parameter	**h1072 = 0**
Window 1 mode-dependent parameter	**h1073 = 0**
Window 1 mode-dependent parameter	**h1074 = 0**
Window 1 address in DPU RAM	0
Window 1 memory window ID	0
Window 2 bottom left x coord	**h1054 = 832**
Window 2 bottom left y coord	**h1055 = 0**
Window 2 x size (centroided pixels)	**h1056 = 64**
Window 2 y size (centroided pixels)	**h1057 = 1024**
Window 2 operation mode	**h1075 = 0**
Window 2 mode-dependent parameter	**h1076 = 0**
Window 2 mode-dependent parameter	**h1077 = 0**
Window 2 mode-dependent parameter	**h1078 = 0**

coninued on next page

(continued)

Description	Params
Window 2 mode-dependent parameter	**h1079 = 0**
Window 2 address in DPU RAM	0
Window 2 memory window ID	0
Window 3 bottom left x coord	**h1058 = 896**
Window 3 bottom left y coord	**h1059 = 0**
Window 3 x size (centroided pixels)	**h1060 = 64**
Window 3 y size (centroided pixels)	**h1061 = 1024**
Window 3 operation mode	**h1080 = 0**
Window 3 mode-dependent parameter	**h1081 = 0**
Window 3 mode-dependent parameter	**h1082 = 0**
Window 3 mode-dependent parameter	**h1083 = 0**
Window 3 mode-dependent parameter	**h1084 = 0**
Window 3 address in DPU RAM	0
Window 3 memory window ID	0
Window 4 bottom left x coord	**h1062 = 960**
Window 4 bottom left y coord	**h1063 = 0**
Window 4 x size (centroided pixels)	**h1064 = 64**
Window 4 y size (centroided pixels)	**h1065 = 1024**
Window 4 operation mode	**h1085 = 0**
Window 4 mode-dependent parameter	**h1086 = 0**
Window 4 mode-dependent parameter	**h1087 = 0**
Window 4 mode-dependent parameter	**h1088 = 0**
Window 4 mode-dependent parameter	**h1089 = 0**
Window 4 address in DPU RAM	0
Window 4 memory window ID	0

Table 41: Flatfield Memory Windows

Description	**Params**
Number of memory windows	**h1159 = 1**
Window 1 bottom left x coord	**h1050 = 0**
Window 1 bottom left y coord	**h1051 = 0**
Window 1 x size (centroided pixels)	**h1052 = 1024**
Window 1 y size (centroided pixels)	**h1053 = 1024**
Window 1 address in DPU RAM	0

9.12 Procedure for the End of Science Data Acquisition Period

At the end of the science data acquisition period of an orbit, care has to be taken that the science data of the last exposure of the last observation are compressed and are made available for downlinking. This can be ensured by forcing the flush compression queue, as follows:

The ICU is in "Science" mode (mode 3).

Description	**tc**	**tm**	**params**	**VxWorks**
Flush the compression queue	**h7255**			`tc_send_command IC_FLUSH_CMPRS`
Wait up to 1000 s for alert DA_EOT_FLUSH_CMPRS and data to come out		**92217**		`tm_wait_for_alert DA_EOT_FLUSH_CMPRS`
Change to ICU mode 2 ("idle")	**h9002**			`tc_mode 2`
Change to ICU mode 5 ("intermediate safe") (housekeeping may be suspended for up to 10 s)	**h9005**			`tc_mode 5`
Wait up to 20 s for a $V_{cathode}$ event packet		**92100**	**h7000="Vcathode reached"**	
Wait up to 20 s for filter wheel event packet		**92600**	**h7010="fw at req pos"**	

The ICU is in "intermediate safe" mode.

9.13 Procedure for Slew

The ICU is in “idle” mode.

Description	**tc**	**tm**	**params**	**VxWorks**
Perform “Flatfielding” (see Section 9.11)				

The ICU is in “idle” mode.

9.14 Procedure for the Preparation for Perigee Passage

The remaining science data in the compressed data buffer should have already have been downloaded (see Section 9.12).

Time tagged "goto safe mode" commands are needed to ensure HV ramp down in case unexpected LOS occurs. This command will be scheduled at the instrument activiation period of an orbit and will be removed as part of the preparation for perigee passage procedure.

The ICU is in "intermediate safe" mode after performing the "End of Science Data Acquisition Period" procedure (see Section 9.12).

Description	**tc**	**tm**	**params**	**VxWorks**
Change to ICU mode 1 ("safe")	**h9001**			`tc_mode 1`
Wait up to 20 s for a V_{mcp1} tm packet		**92100**	**h7000="MCP1 OK"**	
Wait up to 20 s for a V_{mcp23} tm packet		**92100**	**h7000="MCP2,3 OK"**	
Wait up to 20 s for a filter wheel safed tm		**92600**	**h7010="FW @ req pos"**	
If filter wheel has not yet been to datum, wait up to 20 s for another filter wheel event packet		**92600**	**h7010="fw at req pos"**	
Perform "Nominal Switch Off Procedure" (see Section 9.2)				
Spacecraft procedure: remove the MOC's time tagged "goto safe" command				

The ICU is off.

9.15 Post Perigee Procedure

9.15.1 Switch On

Description	tc	tm	params	VxWorks
Perform the "Nominal Switch On" procedure (see Section 9.1)				

The ICU is in "initial" mode.

9.15.2 Memory Dump of ICU's Loadable Code

Send dump memory packets to dump out the ICU's operational mode (loadable) code.

Description	tc	tm	params	VxWorks
Multiple dump memory requests instruction space	**multi h4101**		**h500=*address*** **h510=*length***	
Multiple dump memory requests operand space	**multi h4100**		**h500=*address*** **h510=*length***	

Compare the memory dump report tm packets with the memory load packets. Reload full code if a difference is found (see Section 6.2.6).

9.15.3 Operational Mode

The ICU is in "initial" mode.

Description	tc	tm	params	VxWorks
Mode change to 1 ("full safe")	**h9001**			`tc_mode 1`
Wait up to 20 s				
Check HK		**90000**	**h5405="Operational"**	

The ICU is in "full safe" mode.

9.15.4 Memory Dump of DPU's Loadable Code

Send dump memory packets to dump out the DPU's loadable code.

Description	tc	tm	params	VxWorks
Multiple Dump Bank 0 requests	**multi h4120**		**h500=*address* h510=*length***	
Multiple Dump Bank 3 requests	**multi h4123**		**h500=*address* h510=*length***	
Multiple Dump Bank 5 requests	**multi h4125**		**h500=*address* h510=*length***	
Multiple Dump Bank 7 requests	**multi h4127**		**h500=*address* h510=*length***	

Compare the memory dump report tm packets with the memory load packets. Reload full code if a difference is found (see Section 6.2.6).

9.15.5 Clean Slate

Description	tc	tm	params	VxWorks
Switch to the uploadable code	**h7202**			`tc_load_dpuos`
Wait up to 10 s for DA_DPUOS_READY		**92202**		Expect `''Jim from the Rim''`
Enable red dsp	**h7207**		**h522="Red" h16="Enabled"**	`tc_enbl_dsp red,1`
Wait 10 s				
Enable blue1 dsp	**h7207**		**h522="Blue1" h16="Enabled"**	`tc_enbl_dsp blue1,1`
Wait 10 s				
Enable blue2 dsp	**h7207**		**h522="Blue2" h16="Enabled"**	`tc_enbl_dsp blue2,1`
Wait 10 s				
(Watch out for NCR 94 crash—see Section 11 and Appendix B)				
Select blue2 as the klingon	**h7206**		**h523="blue2"**	`tc_sel_klingon 2`
Initialize DPU	**h7248**			`tc_init_dpu`
Wait up to 25 s for DA_EOT_INIT_DPU event		**92210**		`tm_wait_for_alert DA_EOT_INIT_DPU` ("Dave from the Grave")
Turn on reference frame verbosity (DD_REF)	**h7244**		**h760=3 h16="enable"**	`tc_enbl_verbose 3, 1`

9.15.6 Filter Wheel to Datum

Description	tc	tm	params	VxWorks
Specify next move of the filter wheel to the datum position	**h7608**			`tc_fw_to_datum`

continued on next page

(continued)

Description	tc	tm	params	VxWorks
Command the filter wheel to move	**h5600**			`tc_fw_move`
Wait up to 25 s for a TMPSU event (housekeeping may be suspended for up to 15 s)		**92600**	**h7010="fw @ req pos"**	
Change to ICU mode 2 ("idle")	**h9002**			`tc_mode 2`

9.15.7 Filter Wheel to Blocked

Description	tc	tm	params	VxWorks
Set the filter wheel to blocked	**h7604**		**h280="Blocked"**	`tc_fw_filter 0`
Command the filter wheel to move	**h5600**			`tc_fw_move`
Wait up to 20 s for a TMPSU event (housekeeping may be suspended for up to 15 s)		**92600**	**h7010="fw @ req pos"**	

9.15.8 Science Aux Data Enable

Look Kate ...

Description	tc	tm	params	VxWorks
Enable science aux data	**NOT IN DATABASE: see Fabio**			`tc_enable_sid 0xff`

9.15.9 HV

The following HV ramp-up is for the Prime Channel only.

Description	tc	tm	params	VxWorks
Check HK for temperatures in allowed range and if not, wait		**90000**	**h5135="-8 to $+37$ °C"**	
Transition to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`

continued on next page

(continued)

Description	tc	tm	params	VxWorks
Specify V_{mcp23} ramp parameters	**h7140**		**h150="mcp23" h160=100 h175="off"**	`tc_det_load_hv_ramp_params 3, 100, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 20 s for event		**92100**	**h7000="mcp23 ok"**	
Specify V_{mcp23} ramp parameters	**h7140**		**h150="mcp23" h160=1400 h175="off"**	`tc_det_load_hv_ramp_params 3, 1400, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 90 s for event		**92100**	**h7000="mcp23 ok"**	
Specify V_{mcp1} ramp parameters	**h7140**		**h150="mcp1" h160=550 h175="off"**	`tc_det_load_hv_ramp_params 2, 550, 25, 0`
Wait up to 40 s for event		**92100**	**h7000="mcp1 ok"**	
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Specify $V_{cathode}$ ramp parameters	**h7140**		**h150="cathode" h160=396 h175="off"**	`tc_det_load_hv_ramp_params 1, 396, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 25 s for event		**92100**	**h7000="cathode ok"**	
Specify V_{mcp1} ramp parameters	**h7140**		**h150="mcp1" h160=520 h175="off"**	`tc_det_load_hv_ramp_params 2, 520, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 20 s for event		**92100**	**h7000="mcp1 ok"**	
Specify V_{mcp23} ramp parameters	**h7140**		**h150="mcp23" h160=1700 h175="off"**	`tc_det_load_hv_ramp_params 3, 1700, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 30 s for event		**92100**	**h7000="mcp23 ok"**	

The following HV ramp-up is for the Redundant Channel only.

Note: this should only be used if the prime channel fails.

Description	tc	tm	params	VxWorks
Check HK for temperatures in allowed range and if not, wait		**90000**	**h5135="-8 to $+37$ °C"**	
Transition to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`
Specify V_{mcp23} ramp parameters	**h7140**		**h150="mcp23" h160=100 h175="off"**	`tc_det_load_hv_ramp_params 3, 100, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 10 s for event		**92100**	**h7000="mcp23 ok"**	
Specify V_{mcp23} ramp parameters	**h7140**		**h150="mcp23" h160=1500 h175="off"**	`tc_det_load_hv_ramp_params 3, 1500, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 90 s for event		**92100**	**h7000="mcp23 ok"**	
Specify V_{mcp1} ramp parameters	**h7140**		**h150="mcp1" h160=500 h175="off"**	`tc_det_load_hv_ramp_params 2, 500, 25, 0`

continued on next page

(continued)

Description	tc	tm	params	VxWorks
Wait up to 40 s for event		**92100**	**h7000="mcp1 ok"**	
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Specify $V_{cathode}$ ramp parameters	**h7140**		**h150="cathode" h160=390 h175="off"**	`tc_det_load_hv_ramp_params 1, 390, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 30 s for event		**92100**	**h7000="cathode ok"**	
Specify V_{mcp1} ramp parameters	**h7140**		**h150="mcp1" h160=550 h175="off"**	`tc_det_load_hv_ramp_params 2, 550, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 10 s for event		**92100**	**h7000="mcp1 ok"**	
Specify V_{mcp23} ramp parameters	**h7140**		**h150="mcp23" h160=1850 h175="off"**	`tc_det_load_hv_ramp_params 3, 1850, 25, 0`
Perform ramp	**h5140**			`tc_det_start_hv_ramp`
Wait up to 30 s for event		**92100**	**h7000="mcp23 ok"**	

9.15.10 Pulse Height Distribution

Description	tc	tm	params	VxWorks
Perform Engineering Mode 6 (see Section 10.12) 20 s exposure with filter in the blocked position, a threshold of 12 and flood LEDs at level 4				

9.15.11 LUT Update

This procedure calculates and loads the centroiding boundary lookup table.

Description	tc	tm	params	VxWorks
Perform "Engineering Mode 3: Channel Boundary Data" procedure (see Section 10.9)				
Perform "Engineering Mode 5: Centroiding Confirmation Data" procedure (see Section 10.11)				
Check the output on the ground				

9.15.12 Time Verification

Perform Clock Resynchronisation (see Section 10.1).

Description	tc	tm	params	VxWorks
Send a tc (10, 5)				`tc_enable_time_verification`
Wait up to 2 s for a tm (10, 5)				
Check that the contents of the time management report packet are as expected				

9.15.13 Dave

Description	tc	tm	params	VxWorks
Initialize DPU	**h7248**			`tc_init_dpu`
Wait up to 25 s for DA_EOT_INIT_DPU event		**92210**		`tm_wait_for_alert DA_EOT_INIT_DPU` ("Dave from the Grave")

The ICU is in "idle" mode.

9.15.14 Science Mode

The ICU is in "idle" mode.

Description	tc	tm	params	VxWorks
Change to ICU mode 3 ("science")	**h9003**			`tc_mode 3`

The ICU is in "science mode.

9.16 Procedure for the Preparation for Eclipse

Before transition into eclipse, Optical Monitor has to be switched off by invoking the "Nominal Switch Off Procedure" (see Section 9.2). The remaining data in the compressed data queue buffer have been already sent down ("flushed"—see Section 9.12).

The ICU is in "intermediate safe" mode after performing the "End of Science Data Acquisition Period" procedure (see Section 9.12).

Description	**tc**	**tm**	**params**	**VxWorks**
Perform "Nominal Switch Off Procedure" (see Section 9.2)				

The ICU is off.

9.17 Procedure for Activation After Eclipse

Wait until after the perigee LOS.

The ICU is off.

Description	tc	tm	params	VxWorks
Perform “Post Perigee Procedure” (see Section 9.15)				

The ICU is in “science” mode.

9.18 Pre-apogee Gap Procedure

Normal operation.

9.19 Procedure for Activation After the Apogee Gap

Normal operation.

10 Maintenance Procedures

10.1 Clock Resynchronisation

Description	**tc**	**tm**	**params**	**VxWorks**
Check HK for time sync flag		**90000**	**h5330="inactive"**	
Check time invalid flag in HK		**90000**	**h5335="invalid"**	
Enable time synchronization				`tc_enable_time_synch`
Check HK for time sync flag		**90000**	**h5330="active"**	
Check RBI config register in HK		**90000**	**h5345=1**	
Spacecraft procedure: send bcp3				`bcp3`
Check HK for time sync still enabled		**90000**	**h5330="active"**	
Check RBI config register in HK		**90000**	**h5345=0**	
Add time code				`tc_add_time_code` *time*
Check HK for time sync disabled		**90000**	**h5330="inactive"**	
Check HK for time code as expected		**90000**		
Check HK for time invalid flag		**90000**	**h5335="valid"**	

10.2 Procedure to Change Heater Configuration

The procedure for changing the heater configuration can be used for the heater control test or for contingency. For the heater checkout one would cycle through on/off of all heaters individually. (See XMM-OM/MSSL/ML/0010: User Manual Part 3: Telecommand and Telemetry Specification, Section 2.2.4.4.3, pages 27–28).

The ICU is in mode 1–5 (any of the operational modes).

Description	**tc**	**tm**	**params**	**VxWorks**
Load up heater configuration	One of **h7671** to **h7678**		(see XMM-OM/MSSL/ML/0010: User Manual Part 3: Telecommand and Telemetry Specification, Section 2.2.4.4.3.2)	`tc_heater_function` *fid, param1, param2, param3*
Start heater control again	**h5670**			`tc_heater_start`

10.3 Dichroic Reconfiguration

10.3.1 Selecting the Prime Chain

Note: this is only to be performed if the prime channel fails. It should not be performed as a "test".

The ICU is in any operational mode.

Description	**tc**	**tm**	**params**	**VxWorks**
Change to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Request dichroic move to the prime position	**h7650**		**h370**=−1 **h375=0**	`tc_dm_direction -1,0`
Move dichroic	**h5650**			`tc_dm_move`
Wait for TMPSU event		**92600**	**h7010="dich @ req pos"**	
Examine HK		**90000**	**h5275**=−31	
Return to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`

The ICU is in "idle" mode.

10.3.2 Selecting the Redundant Chain

The ICU is in any operational mode.

Description	**tc**	**tm**	**params**	**VxWorks**
Change to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Request dichroic move to the redundant position	**h7650**		**h370=1 h375=0**	`tc_dm_direction 1,0`
Move dichroic	**h5650**			`tc_dm_move`
Wait for TMPSU event		**92600**	**h7010="dich @ req pos"**	
Examine HK		**90000**	**h5275=32**	
Return to ICU mode 2 ("idle mode")	**h9002**			`tc_mode 2`

The ICU is in "idle" mode.

10.3.3 Check the Chain

Run through most of the commissioning tests (see Section 7).

10.4 Centroid Table Load

Description	tc	tm	params	VxWorks
Load centroid table boundaries	**h7100**		***Table 71***	`det_get_bounds < `*`filename`*` ; tc_det_load_bounds`
Wait up to 120 s for event		**92100**	**h7000=10h**	

Table 71: Centroid Table Boundaries

Description	**Params**	**Initial Value**
Verification	**h10**	"enable"
X Centroid Table Boundary 0	**h20**	-1000
X Centroid Table Boundary 1	**h21**	-750
X Centroid Table Boundary 2	**h22**	-500
X Centroid Table Boundary 3	**h23**	-250
X Centroid Table Boundary 4	**h24**	0
X Centroid Table Boundary 5	**h25**	250
X Centroid Table Boundary 6	**h26**	500
X Centroid Table Boundary 7	**h27**	750
X Centroid Table Boundary 8	**h28**	1000
Y Centroid Table Boundary 0	**h30**	-1000
Y Centroid Table Boundary 1	**h31**	-750
Y Centroid Table Boundary 2	**h32**	-500
Y Centroid Table Boundary 3	**h33**	-250
Y Centroid Table Boundary 4	**h34**	0
Y Centroid Table Boundary 5	**h35**	250
Y Centroid Table Boundary 6	**h36**	500
Y Centroid Table Boundary 7	**h37**	750
Y Centroid Table Boundary 8	**h38**	1000

These are equal bounds. The initial values will be decided later. (See Section 7.)

10.5 Engineering Mode 0: Raw Data of Both DSPs

The ICU is in "idle" mode (2). All DSPs are up.

Description	**tc**	**tm**	**params**	**VxWorks**
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Set frame time	**h7236**		**h710=*frame_time***	`tc_set_frame_time` *frame_time*
Set exposure id	**h7238**		**h530=*exposure_id***	`tc_set_exp_no` *exposure_id*
Send IC_INIT_EXP	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
Load window table	**h7110**		***Table 74***	`det_reset_wind ; det_wind x0, y0, xsize, ysize`
Load windows	**h5110**			`tc_det_load_windows 1`
Wait up to 60 s for event		**92100**	**h7000="win tab load"**	
Enable or diable frame tags	**h7135**		**h16="enabled" or "disabled"**	`tc_det_enbl_frame_tag` *enable / disable*
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *threshold*
Set acquisition mode	**h7130**		**h110="hi res win"**	`tc_det_acq_mode 3`
Start sending events from mic	**h5130**			`tc_det_start_int`
Choose engineering mode 0	**h7254**		**h880="Raw + 2 blue"**	`tc_enbl_eng 1, 0`
Wait up to (*frame_time* + 1) s for end of exposure and complete exposure		**92205** **92209**		`tm_wait_for_alert DA_ENDOF_EXP ; tm_wait_for_alert DA_COMPLETE_EXP`
Wait as long as necessary for enough data (approx. 0–2 hr)				
Reset DPU	**h7201**			`tc_reset_dsp`
Wait up to 12 s for alert		**92201**		Expect "Fred from the Dead"
Run Clean Slate (see Section 6.2.13)				
Change to ICU mode 2 ("idle")	**h9002**			`tc_mode 2`

Table 73: Window Table 256 x 256

Description	**Params**	**Value**	
		Prime	**Redundant**
Verification	h10	"enable"	"enable"
Number of windows	h40	4	4
Window 1 Xlow	h50	52	64
Window 1 Ylow	h51	27	15
Window 1 Xsize	h52	256	256
Window 1 Ysize	h53	256	256

Table 74: General Detector Window Table

Description	**Params**
Verification	abf h10
Number of windows	**h40**
Window 1 Xlow	**h50**
Window 1 Ylow	**h51**
Window 1 Xsize	**h52**
Window 1 Ysize	**h53**
Window 2 Xlow	**h50**
Window 2 Ylow	**h51**
Window 2 Xsize	**h52**
Window 2 Ysize	**h53**
Window 3 Xlow	**h50**
Window 3 Ylow	**h51**
Window 3 Xsize	**h52**
Window 3 Ysize	**h53**
Window 4 Xlow	**h50**
Window 4 Ylow	**h51**
Window 4 Xsize	**h52**
Window 4 Ysize	**h53**

10.6 Reset

Description	tc	tm	params	VxWorks
Spacecraft command: RBI reset cold	**D????**			`<reset`
Wait up to 10 s for boot event, boot exception or boot anomaly packet				
Upload "initial" mode patches using RBI write spacecraft commands				

10.7 Engineering Mode 1: Raw Data of Blue DSP1

The ICU is in "idle" mode (2). All DSPs are up.

Description	**tc**	**tm**	**params**	**VxWorks**
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Initialize DPU	**h7248**			`tc_init_dpu`
Wait up to 25 s for DA_EOT_INIT_DPU event		**92210**		`tm_wait_for_alert DA_EOT_INIT_DPU` ("Dave from the Grave")
Set frame time	**h7236**		**h710=*frame_time***	`tc_set_frame_time` *`frame_time`*
Set exposure id	**h7238**		**h530=*exposure_id***	`tc_set_exp_no` *`exposure_id`*
Send IC_INIT_EXP	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
Load window table	**h7110**		***Table 74***	`det_reset_wind ; det_wind x0, y0, xsize, ysize`
Load windows	**h5110**			`tc_det_load_windows 1`
Wait up to 60 s for event		**92100**	**h7000="win tab load"**	
Ensure frame tags are enabled	**h7135**		**h16="enabled"**	`tc_det_enbl_frame_tag 1`
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *`threshold`*
Set acquisition mode	**h7130**		**h110="hi res win"**	`tc_det_acq_mode 3`
Start sending events from mic	**h5130**			`tc_det_start_int`
Choose engineering mode 1	**h7254**		**h880="Raw + blue 1"**	`tc_enbl_eng 1, 1`
Wait up to (*frame_time* + 1) s for end of exposure and complete exposure		**92205** **92209**		`tm_wait_for_alert DA_ENDOF_EXP ; tm_wait_for_alert DA_COMPLETE_EXP`
Wait as long as necessary for enough data (approx. 0–2 hr)				
Reset DPU	**h7201**			`tc_reset_dsp`
Wait up to 12 s for alert		**92201**		Expect "Fred from the Dead"
Run Clean Slate (see Section 6.2.13)				
Change to ICU mode 2 ("idle")	**h9002**			`tc_mode 2`

10.8 Engineering Mode 2: Raw Data of Blue DSP2

The ICU is in "idle" mode (2). All DSPs are up.

Description	tc	tm	params	VxWorks
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Initialize DPU	**h7248**			`tc_init_dpu`
Wait up to 25 s for DA_EOT_INIT_DPU event		**92210**		`tm_wait_for_alert DA_EOT_INIT_DPU` ("Dave from the Grave")
Set frame time	**h7236**		**h710=*frame_time***	`tc_set_frame_time` *frame_time*
Set exposure id	**h7238**		**h530=*exposure_id***	`tc_set_exp_no` *exposure_id*
Send IC_INIT_EXP	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
Load window table	**h7110**		***Table 74***	`det_reset_wind ; det_wind x0, y0, xsize, ysize`
Load windows	**h5110**			`tc_det_load_windows 1`
Wait up to 60 s for event		**92100**	**h7000="win tab load"**	
Ensure frame tags are enabled	**h7135**		**h16="enabled"**	`tc_det_enbl_frame_tag 1`
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *threshold*
Set acquisition mode	**h7130**		**h110="hi res win"**	`tc_det_acq_mode 3`
Start sending events from mic	**h5130**			`tc_det_start_int`
Choose engineering mode 2	**h7254**		**h880="Raw + blue 2"**	`tc_enbl_eng 1, 2`
Wait up to (*frame_time* + 1) s for end of exposure and complete exposure		**92205** **92209**		`tm_wait_for_alert DA_ENDOF_EXP ; tm_wait_for_alert DA_COMPLETE_EXP`
Wait as long as necessary for enough data (approx. 0–2 hr)				
Reset DPU	**h7201**			`tc_reset_dsp`
Wait up to 12 s for alert		**92201**		Expect "Fred from the Dead"
Run Clean Slate (see Section 6.2.13)				
Change to ICU mode 2 ("idle")	**h9002**			`tc_mode 2`

10.9 Engineering Mode 3: Channel Boundary Data

This loads the Centroiding Lookup Table.

The ICU is in "idle" mode (2).

Description	**tc**	**tm**	**params**	**VxWorks**
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Initialize DPU	**h7248**			`tc_init_dpu`
Move the filter wheel to blocked	**h7604**		**h280="Blocked"**	`tc_fw_filter 0`
Move the filter wheel (housekeeping may be suspended for up to 10 s)	**h5600**			`tc_fw_move`
Wait up to 20 s for TMPSU event report		**92600**	**h7010="fw @ req pos"**	
Turn on flood LEDs to level 4	**h7134**		**h1040=4**	`tc_det_flood_led_current 4`
Wait up to 25 s for DA_EOT_INIT_DPU event		**92210**		`tm_wait_for_alert DA_EOT_INIT_DPU` ("Dave from the Grave")
Set frame time to 300 s	**h7236**		**h710=*frame_time***	`tc_set_frame_time _time`
Set exposure id	**h7238**		**h530=*exposure_id***	`tc_set_exp_no` *`exposure_id`*
Send IC_INIT_EXP	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
(Enable M,N image dump, if wanted)	**h7244**		**h760="eng data"** **h16="enable"**	tc_enbl_verbose 4,1
Load window table	**h7110**		***Table 73***	`det_reset_wind ; det_wind x0, y0, xsize, ysize`
Load windows	**h5110**			`tc_det_load_windows 1`
Wait up to 60 s for event		**92100**	**h7000="win tab load"**	
Ensure frame tags are disabled	**h7135**		**h16="disabled"**	`tc_det_enbl_frame_tag 0`
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *`threshold`*
Set acquisition mode	**h7130**		**h110="eng x m/n"**	`tc_det_acq_mode 4`
Start sending events from mic	**h5130**			`tc_det_start_int`
Choose engineering mode 3	**h7254**		**h880="Centroiding"**	`tc_enbl_eng 1, 3`
Wait up to 301 s for end of exposure		**92205**		`tm_wait_for_alert DA_ENDOF_EXP`
Wait up to 79 s for complete exposure		**92209**		`tm_wait_for_alert DA_COMPLETE_EXP` and end of data tm packets
Wait up to 0 s for telemetry to finish		Multi- **97322** **92207**		

continued on next page

(continued)

Description	**tc**	**tm**	**params**	**VxWorks**
Turn off flood LEDs	**h7134**		**h1040=0**	`tc_det_flood_led_current 0`
Change to ICU mode 2 (“idle”)	**h9002**			`tc_mode 2`

10.10 Engineering Mode 4: Full Frame, High Resolution

The ICU is in "idle" mode (2).

Description	tc	tm	params	VxWorks
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Initialize DPU	**h7248**			`tc_init_dpu`
Wait up to 25 s for DA_EOT_INIT_DPU event		**92210**		`tm_wait_for_alert DA_EOT_INIT_DPU` ("Dave from the Grave")
Set frame time in units of 1/1024 s	**h7236**		**h710=*frame_time***	`tc_set_frame_time` *`frame_time`*
Set exposure id	**h7238**		**h530=*exposure_id***	`tc_set_exp_no` *`exposure_id`*
Send IC_INIT_EXP	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
Load window table	**h7110**		***Table 73***	`det_reset_wind ; det_wind x0, y0, xsize, ysize`
Load windows	**h5110**			`tc_det_load_windows 1`
Wait up to 60 s for event		**92100**	**h7000="win tab load"**	
Ensure frame tags are disabled	**h7135**		**h16="disabled"**	`tc_det_enbl_frame_tag 0`
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *`threshold`*
Set acquisition mode	**h7130**		**h110="hi res win"**	`tc_det_acq_mode 3`
Start sending events from mic	**h5130**			`tc_det_start_int`
Choose engineering mode 4	**h7254**		**h880="Full Hi Res"**	`tc_enbl_eng 1, 4`
Wait up to *frame_time + 1* s for end of exposure		**92205**		`tm_wait_for_alert DA_ENDOF_EXP`
Wait up to 4800 s for complete exposure		**92209**		`tm_wait_for_alert DA_COMPLETE_EXP`
Wait up to 1000 s for eng data telemetry to finish		Multi- **97322** **92207**		
Change to ICU mode 2 ("idle")	**h9002**			`tc_mode 2`

Table 80: Window Table

Description	Params
Verification	**h10**
Number of windows	**h40**
Window 1 Xlow	**h50**
Window 1 Ylow	**h51**
Window 1 Xsize	**h52**
Window 1 Ysize	**h53**
Window 2 Xlow	**h54**
Window 2 Ylow	**h55**
Window 2 Xsize	**h56**
Window 2 Ysize	**h57**
.	.
.	.
.	.
Window 14 Xlow	**h102**
Window 14 Ylow	**h103**
Window 14 Xsize	**h104**
Window 14 Ysize	**h105**
Window 15 Xlow	**h106**
Window 15 Ylow	**h107**
Window 15 Xsize	**h108**
Window 15 Ysize	**h109**

10.11 Engineering Mode 5: Centroiding Confirmation Data

The ICU is in "idle" mode (2).

Description	**tc**	**tm**	**params**	**VxWorks**
Change to ICU mode 4 ("engineering mode")	**h9004**			`tc_mode 4`
Initialize DPU	**h7248**			`tc_init_dpu`
Wait up to 25 s for DA_EOT_INIT_DPU event		**92210**		`tm_wait_for_alert DA_EOT_INIT_DPU` ("Dave from the Grave")
Move the filter wheel to blocked	**h7604**		**h280="Blocked"**	`tc_fw_filter 0`
Move the filter wheel (housekeeping may be suspended for up to 10 s)	**h5600**			`tc_fw_move`
Wait up to 20 s for TMPSU event report		**92600**	**h7010="fw @ req pos"**	
Turn on flood LEDs to level 4	**h7134**		**h1040=4**	`tc_det_flood_led_current 4`
Set the DPU frame time to (60 × 1024) in units of 1/1024 s	**h7236**		**h710=*frame_time***	`tc_set_frame_time` *`frame_time`*
Set exposure id	**h7238**		**h530=*exposure_id***	`tc_set_exp_no` *`exposure_id`*
Send IC_INIT_EXP	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
Load window table	**h7110**		***Table 73***	`det_reset_wind ; det_wind x0, y0, xsize, ysize`
Load windows	**h5110**			`tc_det_load_windows 1`
Wait up to 60 s for event		**92100**	**h7000="win tab load"**	
Ensure frame tags are disabled	**h7135**		**h16="disabled"**	`tc_det_enbl_frame_tag 0`
Set threshold	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *`threshold`*
Set acquisition mode	**h7130**		**h110="hi res win"**	`tc_det_acq_mode 3`
Start sending events from mic	**h5130**			`tc_det_start_int`
Choose engineering mode 5	**h7254**		**h880="Centr Conf"**	`tc_enbl_eng 1, 5`
Wait up to *frame_time*+1 s for end of exposure		**92205**		`tm_wait_for_alert DA_ENDOF_EXP`
Wait up to 150 s for complete exposure		**92209**		`tm_wait_for_alert DA_COMPLETE_EXP`
Wait up to 20 s for eng data telemetry to finish		Multi- **97322** **92207**		
Turn off flood LEDs	**h7134**		**h1040=0**	`tc_det_flood_led_current 0`
Change to ICU mode 2 ("idle")	**h9002**			`tc_mode 2`

10.12 Engineering Mode 6: Intensifier Characteristics

The ICU is in “idle” mode (2).

Description	**tc**	**tm**	**params**	**VxWorks**
Change to ICU mode 4 (“engineering mode”)	**h9004**			`tc_mode 4`
Initialize DPU	**h7248**			`tc_init_dpu`
Wait up to 25 s for DA_EOT_INIT_DPU event		**92210**		`tm_wait_for_alert DA_EOT_INIT_DPU` (“Dave from the Grave”)
Move the filter wheel to blocked	**h7604**		**h280=“Blocked”**	`tc_fw_filter 0`
Move the filter wheel (housekeeping may be suspended for up to 10 s)	**h5600**			`tc_fw_move`
Wait up to 20 s for TMPSU event report		**92600**	**h7010=“fw @ req pos”**	
Turn on flood LEDs to level 4	**h7134**		**h1040=4**	`tc_det_flood_led_current 4`
Set the DPU frame time (in units of 1/1024 s)	**h7236**		**h710=*frame_time***	`tc_set_frame_time` *frame_time*
Set exposure id	**h7238**		**h530=*exposure_id***	`tc_set_exp_no` *exposure_id*
Send IC_INIT_EXP	**h7249**			`tc_send_command IC_INIT_EXP`
Wait 3 s				
Load window table	**h7110**		***Table 73***	`det_reset_wind ; det_wind x0, y0, xsize, ysize`
Load windows	**h5110**			`tc_det_load_windows 1`
Wait up to 60 s for event		**92100**	**h7000=“win tab load”**	
Ensure frame tags are disabled	**h7135**		**h16=“disabled”**	`tc_det_enbl_frame_tag 0`
Set threshold to 12	**h7131**		**h120=*threshold***	`tc_det_event_threshold` *threshold*
Set acquisition mode	**h7130**		**h110=“eng evt ht”**	`tc_det_acq_mode 6`
Start sending events from mic	**h5130**			`tc_det_start_int`
Choose engineering mode 6	**h7254**		**h880=“Intensifier”**	`tc_enbl_eng 1, 6`
Wait up to 21 s for end of exposure		**92205**		`tm_wait_for_alert DA_ENDOF_EXP`
Wait up to 20 s for complete exposure		**92209**		`tm_wait_for_alert DA_COMPLETE_EXP`
Wait up to 20 s for telemetry to finish		Multi- **97322** **92207**		
Turn off flood LEDs	**h7134**		**h1040=0**	`tc_det_flood_led_current 0`
Change to ICU mode 2 (“idle”)	**h9002**			`tc_mode 2`

11 Error recovery

If an error occurs in the running of the ICU of DPU code, the following procedures should be followed to help understand the problem and make the instrument safe.

If the OM needs to be reset, the "initial" mode patches should be uploaded after the reset command (see Section 10.6).

11.1 Unsuccessful Command Acceptance (3,2)

Check the telecommand. Check that the ICU and DPU are in the appropriate mode for that command.

11.2 Unsuccessful Command Execution (3,3)

11.3 Non-Periodic Housekeeping: Non-DPU Exception Reports (4,2)

11.3.1 Busy

Check the time since the last similar command.

11.3.2 Ada Exceptions

Perform the ICU debug procedure (see Section 12) then "Safe" (reset: see Section 9.3) and follow the "Post Perigee Procedure" (see Section 9.15).

11.3.3 Bootstrap Report

11.4 Non-Periodic Housekeeping: DPU Exception Reports (4,2)

11.5 Major Anomaly Reports

11.6 HK Values out of Range

```
Here is the database information for the DP_FAQ telemetry packet type
and hard and soft limits for relevant TM parameters with associated
recovery actions.

This is in response to both parts of action number 2.09 set at the
OM Operations Meeting #2 at VILSPA 27-28 Jan 99.

Some details, such as the recovery actions for some thermistor warnings
and alarms, are still being worked out but we should be able to get you
details either by e-mail of fax by Monday 15/2/99.

I'm pretty certain that you will be in touch with many questions :)

Cheers,

Matt W.


1) DP_FAQ parameters.
---------------------

Telemetry packet TPN 97402, name "Field Acq Data".


Byte Sbit Width Pref Description Value Fix/Var
----    ----    -----   ----    -----------     -----   -------

4 0 8 FIX (Datatype msb) 162 F
5 0 8 FIX (Datatype lsb) 2 F
6 0 16 FIX (Word count) 145 F
8 0 8 FIX (Sync word msb) 243 F
9 0 8 FIX (Sync word lsb) 82 F
10 0 32 H9200 Exposure Id -V
14 0 32 H9205 Iterations -V
18 0 32 H9210 X command -V
22 0 32 H9215 Y command -V
26 0 32 H9220 dX -V
30 0 32 H9225 dY -V
34 0 32 H9230 sin theta -V
38 0 32 H9235 ngs up -V
42 0 32 H9240 ngs found -V
46 0 32 H9245 gs x0 -V
50 0 32 H9250 gs y0 -V
```

```
54 0 32 H9255 gs x1 -V
58 0 32 H9260 gs y1 -V
. . . . . . .
. . . . . . .
. . . . . . .
158 0 32 H9385 gs x14 -V
162 0 32 H9390 gs y14 -V
166 0 32 H9395 gs x15 -V
170 0 32 H9400 gs y15 -V
174 0 32 H9405 ref star x0 -V
178 0 32 H9410 ref star y0 -V
182 0 32 H9415 ref star x1 -V
186 0 32 H9420 ref star y1 -V
. . . . . . .
. . . . . . .
. . . . . . .
286 0 32 H9545 ref star x14 -V
290 0 32 H9550 ref star y14 -V
294 0 32 H9555 ref star x15 -V
398 0 32 H9560 ref star y15 -V

The PTC/PFC for the 32 bit parameters is 3,14.

Fields for bytes 1 to 3 are already in the database.



2) Condition parameters.
------------------------

A) The value in the TM_COND_MODE field for parameters X9010 and X9020 need to be
   swapped, ie, the mode for X9010 needs to read "H5400 = 1" and for X9202 it
   should read "H5400 = 0".


B) New condition parameters for the secondary rail currents.

   PREF PARA_NAME DESCRIPTION TM_COND_MODE
   ----  --------------------------------

   X9060 PRI CAMON Camera ctl started H5230 = 1 AND
& primary channel H5400 = 1

   X9070 PRI CAMOFF Camera ctl stopped H5230 = 0 AND
& primary channel H5400 = 1 AND
& secondaries on H5255 = 1
```

```
   X9080 RED CAMON Camera ctl started H5230 = 1 AND
& redundant channel H5400 = 0

   X9090 RED CAMOFF Camera ctl stopped H5230 = 0 AND
& redundant channel H5400 = 0 AND
& secondaries on H5255 = 1


C) New condition parameters for the low voltage monitors.

   PREF PARA_NAME DESCRIPTION TM_COND_MODE
   ----  --------------------------------

   X9100 SECON Secondaries on H5255 = 1

   X9110 SECOFF Secondaries off H5255 = 0


D) New condition parameters for the high voltage.

   PREF PARA_NAME DESCRIPTION TM_COND_MODE
   ----  --------------------------------

   X9120 PRI HV ON HV enabled H5145 = 1 AND
& prime channel H5400 = 1

   X9130 RED HV ON HV enabled H5145 = 1 AND
& redundant channel H5400 = 0



3) Warnings and alarms.
-----------------------

A) Warnings and alarms for the secondary rail currents (in milliamps).


   PREF   Name   Cond. Warn Low  Warn High  Alarm Low  Alarm High
   ----   ----  -------------  ---------  ---------  ----------

   H5280  Current +25 SV  X9060    18.00      23.00  16.00      25.00
   H5280  Current +25 SV  X9070    18.00      22.00  16.00      24.00
   H5280  Current +25 SV  X9080    19.00      24.00  17.00      26.00
   H5280  Current +25 SV  X9090    17.00      22.00  15.00      24.00
```

H5285	Current +15 SV	X9060	44.00	54.00	39.00	59.00
H5285	Current +15 SV	X9070	44.00	54.00	39.00	59.00
H5285	Current +15 SV	X9080	42.00	52.00	37.00	56.00
H5285	Current +15 SV	X9090	41.00	50.00	36.00	55.00
H5290	Current +11 SV	X9060	149.00	183.00	133.00	199.00
H5290	Current +11 SV	X9070	90.00	96.00	87.00	99.00
H5290	Current +11 SV	X9080	153.00	188.00	136.00	205.00
H5290	Current +11 SV	X9090	87.00	107.00	77.00	116.00
H5295	Current +5.3 SV	X9060	797.00	975.00	709.00	1064.00
H5295	Current +5.3 SV	X9070	442.00	541.00	393.00	590.00
H5295	Current +5.3 SV	X9080	821.00	1005.00	730.00	1096.00
H5295	Current +5.3 SV	X9090	440.00	539.00	391.00	588.00
H5300	Current -5.3 SV	X9060	432.00	528.00	384.00	576.00
H5300	Current -5.3 SV	X9070	414.00	506.00	368.00	552.00
H5300	Current -5.3 SV	X9080	407.00	498.00	362.00	543.00
H5300	Current -5.3 SV	X9090	405.00	495.00	360.00	540.00
H5305	Current -15 SV	X9060	49.00	60.00	43.00	65.00
H5305	Current -15 SV	X9070	49.00	60.00	43.00	65.00
H5305	Current -15 SV	X9080	47.00	58.00	41.00	63.00
H5305	Current -15 SV	X9090	47.00	58.00	41.00	63.00
H5310	Current +28 UV	X9060	5.00	6.00	4.00	7.00
H5310	Current +28 UV	X9070	5.00	6.00	4.00	7.00
H5310	Current +28 UV	X9080	5.00	6.00	4.00	7.00
H5310	Current +28 UV	X9090	5.00	6.00	4.00	7.00
H5315	Current +5 UV	X9060	100.00	123.00	90.00	134.00
H5315	Current +5 UV	X9070	96.00	117.00	86.00	128.00
H5315	Current +5 UV	X9080	108.00	132.00	96.00	144.00
H5315	Current +5 UV	X9090	108.00	132.00	96.00	144.00

B) Warnings and alarms for the low voltage monitors (in volts).

PREF	Name	Cond.	Warn Low	Warn High	Alarm Low	Alarm High
H5170	LVM +5V	X9100	4.75	5.75	4.00	6.50
H5175	LVM +15V	X9100	13.50	16.50	12.00	18.00
H5180	LVM -15V	X9100	-14.00	-17.00	-12.00	-18.50
H5185	LVM 1.23V Ref	X9100	1.50	1.75	1.00	2.00

C) Warnings and alarms for the thermistors (in degrees centigrade).

PREF	Name	Cond.	Warn Low	Warn High	Alarm Low	Alarm High
H5105	BPE thermistor	X9100	-10.00	40.00	-20.00	50.00
H5135	CCD Thermistor	X9100	-4.00	37.00	-8.00	42.00
H5140	Ref A Thermistor	X9100	18.00	21.00	17.00	22.00
H5110	Ref B Thermistor	X9100	18.00	21.00	17.00	22.00
H5120	Main Thermistor	X9100	18.00	21.00	17.00	22.00
H5115	Ref C Thermistor	X9100	NC	NC	NC	NC
H5125	Fwd 1 Thermistor	X9100	17.00	22.00	16.00	23.00
H5130	Fwd 2 Thermistor	X9100	17.00	22.00	16.00	23.00

Note: Ref A, Ref B and Main thermistors are averaged together to control the heater settings. The actual values for the warnings and alarms will have to be reviewed.
NC = No Connection.

D) Warnings and alarms for the high voltages.

The warnings and alarms for these parameters (PREFs H5155, H5160 and H5165) pose some problems for entry in the database because they can be set to variable levels. I am unsure how to enter limit checks for variable parameters. The only solution I can think of is to enter various working nominal levels expected and set the limits according to that.

For the prime channel (where nom is the expected operating voltage) :

PREF	Name	Nom	Cond.	Warn Low	Warn High	Alarm Low	Alarm High
H5155	HVM Vmcp1	550	X9120	548.00	558.00	541.00	566.00

```
   H5155  HVM Vmcp1 520   X9120   519.00     529.00     511.00      536.00

   H5160  HVM Vmcp23   1400   X9120  1384.00    1423.00    1354.00     1451.00
   H5160  HVM Vmcp23   1700   X9120  1676.00    1716.00    1647.00     1745.00

   H5165  HVM Vcath   2   X9120      N/A      4.00       N/A       5.00
   H5165  HVM Vcath 396   X9120   395.00     397.00     393.00      399.00


   For the redundant channel (where nom is the expected operating voltage) :

   PREF   Name Nom   Cond. Warn Low  Warn High  Alarm Low  Alarm High
   ----   -------  -----  --------  ---------  ---------  ----------

   H5155  HVM Vmcp1 500  X9130    494.00     504.00     487.00      512.00
   H5155  HVM Vmcp1 550  X9130    546.00     556.00     538.00      563.00

   H5160  HVM Vmcp23   1500  X9130   1481.00    1521.00    1452.00     1550.00
   H5160  HVM Vmcp23   1850  X9130   1833.00    1872.00    1803.00     1901.00

   H5165  HVM Vcath   2  X9130       N/A      4.00       N/A       5.00
   H5165  HVM Vcath 390  X9130    389.00     392.00     387.00      393.00

Note: the low warn and alarm are designated N/A as it is not possible to resolve
      such values.



4) Recovery actions.
--------------------

A) Secondary currents.

Low alarm  : If enabled, turn off high voltages in the correct order
     Turn off secondaries (tc H6690)

Low warn   : Monitor closely

High warn  : Monitor closely

High alarm : If enabled, turn off high voltages in the correct order
     Turn off secondaries (tc H6690)

B) Low voltage currents.

Low alarm  : If enabled, turn off high voltages in the correct order
```

```
     Turn off secondaries (tc H6690)

Low warn   : Monitor closely

High warn  : Monitor closely

High alarm : If enabled, turn off high voltages in the correct order
     Turn off secondaries (tc H6690)

C) Thermistors.

   a) BPE and CCD thermistors.

Low alarm  : Monitor closely

Low warn   : Monitor closely

High warn  : Monitor closely

High alarm : Stop detector integration (tc H6130)
     If enabled, turn off high voltage in the correct order
     Stop heater switching (tc H6670)
     Start contingency heater control (tc H5660)
     Uplink heater configuration (tc H7660, H380 = "ONONOFFOFF")
     Turn off secondaries (tc H6690)


   b) Ref A, Ref B, Main thermistors.

Actions on warnings and alarms are still being determined.

   c) Forward 1, forward 2 thermistors.

Actions on warnings and alarms are still being determined.


D) High voltages.

Low alarm  : When ramping, hold and observe

Low warn   : Monitor closely

High warn  : Monitor closely

High alarm : Command instrument to full safe mode (tc H9001)
```

```
In the event of any alarm perform the recovery procedure and
consult the PI institute as to the next course of action.
```

11.7 Data Corruption

Check the OM telemetry rate. Perform the ICU and DPU debug procedures (see Section 12), abort the current exposure and continue with the next.

11.8 Sudden DPU death

Perform the DPU debug procedure (see Section 12). Perform "Clean Slate" (see Section 6.2.13).

11.9 DPU Status Exposure Number Crazy

Check the OM telemetry rate. Perform the ICU and DPU debug procedures (see Section 12), abort the current exposure and continue with the next.

11.10 More ??

11.11 Cheng's error recovery procedures

```
Summary of Error Handling for DPU in Response to DPU Exception Alerts
======================================================================

Cheng Ho,
Los Alamos National Lab
v.1.0 March 27, 1999

1. Summary of existing DPU error (Exception) alerts and Handling procedure

1.1.

#define DA_CLK_SYNCH_ERROR        0XA580

After the DPU is rebooted and DPUOS reloaded, ICU should issue an
IC_SYNCH_CLK command. This first synchronization command should not
generate any error.  After this initial CLK synchronization, further
issuance of IC_SYNCH_CLK will generate a DA_CLK_SYNCH_ERROR within 10
seconds. This is normal.

If a DA_CLK_SYNCH_ERROR occurs not as a response to IC_CLK_SYNCH. If
the alert indicates a loss of time tick, the DPU code may some fine
tuning.

Normal response: Do nothing. The DPU operation should not be
affected. Note the exposure ID, occurrence and subsequent DPU
behaviour. Contact DPU software maintainer for a post facto DPU
diagnostics.  If the alert indicates a gain of time tick on the DPU by
a small number, e.g. 1 or 2, contact the DPU hardware/software
maintainer immediately.

1.2.

#define DA_PROG_RAM_ERROR         0XA581

Do nothing. This alert indicates that DPU code scrubbing has been
performed by the red DSP.

1.3.

#define DA_BOTTOM_WHITEDSP        0XA582

This alert should be followed by a spontaneous DPU reset indicated by
```

an unsheduled DA_DPU_BOOT_READY. Perform DPUPM procedure described in the Appendix and resume OM operation.

1.4.

```
#define DA_CID_TOO_LARGE          0XA583
```

Check the validity of the most recent uploaded ICU to DPU command. That one has not taken on the DPU, reset the DPU and resume OM operation.

1.5.

```
#define DA_BAD_CMD                0XA584
```

Check the vaildity of the most recent uploaded ICU to DPU command. That one has not taken on the DPU, reset the DPU and resume OM operation.

1.6.

```
#define DA_DPU_ABORTING           0XA585
```

This is a placeholder. This indicates that the DPU is in a limbo state. If this is the result of a ground uploaded DPU command that is not part of the standard procedure,, reset DPU completely and proceed. If this occurs in the normal course of OM operation, perform a DPUPMM process then resume OM operation.

1.7.

```
#define DA_BAD_BLOCK              0XA586
```

This alert indicates that DPU has received a bad block from the SSI interface. If the DPU does not generate any further alert, continue with OM operation. Expect the DPU operation to generate erratic results for the current exposure.

1.8.

```
#define DA_ZERO_LENGTH_DQ         0XA587
```

This alert indicates that DPU is attempting to send out a data et of length 0. If the DPU does not generate any further alert, continue with OM operation. Expect the DPU operation to generate erratic results for the current exposure.

1.9.

```
#define DA_PROG_RAM_MERROR       0XA588
```

This alert indicates the detection of an unrecoverable program error in the DPU. It should be preceded by a DA_RED_E_ALERT with an error condition >=0x0100. Reset the DPU and perform a DPUPM. Reload the DPU codes and follow the standard post-code-load procedures. At a later stage, utility might be developed to upload only the culprit block.

1.10.

```
#define DA_SCI_RCVR_ERROR        0XA590
```

Not used.

1.11.

```
#define DA_SCI_HBEAT_ERROR       0XA591
```

This alert could occur under rare occasion at the initial startup of the DPU. If this alert occurs only once or twice, expecially during startup, do nothing and proceed with DPU operation. If this is a alert repeats more than 5 times in 10 seconds indicating the same processor (0=blue1, 2=lue2, 1/3=red), the culprit processor is not operating properly. Reset the DPU and perform the DPUPMM. Resume OM operation afterwards.

1.12.

```
#define DA_RED_E_ALERT           0xA592
```

This alerts carries a 16 bit error word. See ICU-DPU protocal for definition. If single bit error is indicated, do nothing. If multi-bit error is indicated, continue DPU operation until the exposure complete and the A588 alerts appears, then reset DPU and perform DPUPMM. If the DPU spontaneous reset before the A588 appears, reset DPU and perform DPUPMM.

1.13.

```
#define DA_RED_S_ALERT           0xA593
```

This alert is not being used. This alert could occur during start up if the DPU was reset from a strange state. Under such circumstance, do nothing.

1.14.

```
#define DA_BLUE1_E_ALERT          0xA594
```

This alert is not being used unless the bluedsp bad address (BADD) compilation flag is turned on and code loaded with this debugging procedure. Be aware of the code configuration. If BADD is set, consult the DPU code maintainer. If the BADD is not set in the blue codes, this alert could occur during start up if the DPU was reset from a strange state. Under such circumstance, do nothing.

1.15.

```
#define DA_BLUE1_S_ALERT          0xA595
```

This alert is not being used. This alert could occur during start up if the DPU was reset from a strange state. Under such circumstance, do nothing.

1.16.

```
#define DA_BLUE2_E_ALERT          0xA596
```

This alert is not being used unless the bluedsp bad address (BADD) compilation flag is turned on and code loaded with this debugging procedure. Be aware of the code configuration. If BADD is set, consult the DPU code maintainer. If the BADD is not set in the blue codes, this alert could occur during start up if the DPU was reset from a strange state. Under such circumstance, do nothing.

1.17.

```
#define DA_BLUE2_S_ALERT          0xA597
```

This alert is not being used. This alert could occur during start up if the DPU was reset from a strange state. Under such circumstance, do nothing.

1.18.

```
#define DA_GADE_RPI               0XA5A0
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU

operation due to pathological field condition (photon counts too low).
Note the occurence and do nothing.

1.19.

#define DA_GADE_RPIA_L 0XA5A1

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.20.

#define DA_GADE_RPIA_H 0XA5A2

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.21.

#define DA_GADE_WPI 0XA5A3

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.22.

#define DA_GADE_WPIA_L 0XA5A4

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.23.

#define DA_GADE_WPIA_H 0XA5A5

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low).

Note the occurence and do nothing.

1.24.

```
#define DA_GADE_RBI               0XA5A6
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.25.

```
#define DA_GADE_RBIA_L            0XA5A7
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.26.

```
#define DA_GADE_RBIA_H            0XA5A8
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.27.

```
#define DA_GADE_WBI               0XA5A9
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.28.

```
#define DA_GADE_WBIA_L            0XA5AA
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.29.

```
#define DA_GADE_WBIA_H           0XA5AB
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.30.

```
#define DA_GADE_RSI              0XA5AC
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.31.

```
#define DA_GADE_RSIA_L           0XA5AD
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.32.

```
#define DA_GADE_RSIA_H           0XA5AE
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.33.

```
#define DA_GADE_WSI              0XA5AF
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.34.

```
#define DA_GADE_WSIA_L           0XA5B0
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.35.

```
#define DA_GADE_WSIA_H           0XA5B1
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low). Note the occurence and do nothing.

1.36.

```
#define DA_GADE_RANGE_ERROE      0XA5B2
```

This is a global memory access error trap. An illegal global memory access has been requested, typically as a result of erratic DPU operation due to pathological field condition (photon counts too low or star image quality too bad). Note the occurence and do nothing.

1.37.

```
#define         DA_ZERO_CNT_CALBGD       0XA5FF
```

Do nothing. This alert is usually associated with very low count rate into the DPU. Double check the OM operation. If the filter is narrow and the FOV is small, this is not unexpected. Check the health indicators of the MIC detector (high voltage and current reading) and the BPE.

1.38.

```
#define         DA_NO_GOOD_GS            0XA5D7
```

The DPU cannot find any good guide stars to track on. The DPU is likely to continue the exposure with the tracking function turned on. In this case, do nothing. Mark the exposure as such as an advisory for the observer.

1.39.

```
#define         DA_RSTR_BAD_TASK_ID       0XA5FC
```

This alert is generated when the task id was changed during the operation of the su_restore_stars. Review the operation procedure. If this change in DPU task ID is issued from the operation procedure, modify the procedure. Presumably a established procedure should not generate this condition. Under such circumstance, perform a DPU reset and DPUPMM, then proceed with OM operation.

2. Summary of undocumented DPU alerts blocked out by compliation flag or normal parameter setting

The following alerts are present in the DPU codes that are currently blocked off by compilation flag or no used under normal condition. The presence of these alerts are very context dependent. The interpretation and handling of these alerts should be done in conjunction with the code developer/maintainer.

		Comp. or control variable	default
cwhite.c:	0xA5F1	DEBUG_LOAD_CODE==1	0
cwhite.c:	0xA5F2	FEC_UPLINK_TEST==1	0
cwhite.c:	0xA5F0	DEBUG_LOAD_CODE==1	0
su_calbgd.c:	0xA5F0	keep_a_log & 0x8000	0x0001
su_calbgd.c:	0xA5F1	keep_a_log & 0x8000	0x0001
su_calbgd.c:	0xA5F2	keep_a_log & 0x8000	0x0001
su_calbgd.c:	0xA5FE	keep_a_log & 0x8000	0x0001
su_caldrft.c:	0xA5E0	keep_a_log & 0x0200	0x0001
su_caldrft.c:	0xA532	DEBUG_CALDRFT==1	0
su_caldrft.c:	0xA537	DEBUG_CALDRFT==2	0
su_compress.c:	0xA5F1	DPU_DEBUG==10	0
su_compress.c:	0xA5F0	DPU_DEBUG==9	0
su_compress.c:	0xA5E1	DPU_DEBUG==9	0
su_compress.c:	0xA5E2	DPU_DEBUG==9	0
su_compress.c:	0xA5F3	DPU_DEBUG==9	0
su_compress.c:	0xA5E6	DPU_DEBUG==1 or 19	0
su_compress.c:	0xA5E3	DPU_DEBUG==1 or 19	0
su_compress.c:	0xA5Ea	DPU_DEBUG==1 or 19	0
su_compress.c:	0xA5C4	DPU_DEBUG==3	0
su_compress.c:	0xA5C1	DPU_DEBUG==13	0
su_compress.c:	0xA5Cb	DPU_DEBUG==3	0
su_compress.c:	0xA5C0	DPU_DEBUG==3	0
su_compress.c:	0xA5C2	DPU_DEBUG==13	0
su_compress.c:	0xA5F0	DPU_DEBUG==1	0

```
su_compress.c:    0xA5E4   DPU_DEBUG==11             0
su_compress.c:    0xA5E5   DPU_DEBUG==11             0
su_compress.c:    0xA5E5   DPU_DEBUG==1              0
su_compress.c:    0xA5E5   DPU_DEBUG==2              0
su_compress.c:    0xA5E9   DPU_DEBUG==1              0
su_deliverdata.c: 0xA5E2   keep_a_log & 0x2000       0x0001
su_deliverdata.c: 0xA5E3   keep_a_log & 0x2000       0x0001
su_deliverdata.c: 0xA5d8   keep_a_log & 0x2000       0x0001
su_deliverdata.c: 0xA5d9   keep_a_log & 0x2000       0x0001
su_dowindow.c:    0xA5c9   keep_a_log & 0x1000       0x0001
su_dowindow.c:    0xA5F0   DEBUG_DOWINDOW==1         0
su_dowindow.c:    0xA5F5   DEBUG_DOWINDOW==1         0
su_engineering.c: 0xA5E0
su_engineering.c: 0xA5E1
su_engineering.c: 0xA5E3
su_getref.c:      0xA5D1
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C2   STRESS_TEST==1            0
su_initialize.c:  0xA5C3   STRESS_TEST==1            0
su_initialize.c:  0xA5C4   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C2   STRESS_TEST==1            0
su_initialize.c:  0xA5C3   STRESS_TEST==1            0
su_initialize.c:  0xA5C4   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C2   STRESS_TEST==1            0
su_initialize.c:  0xA5C3   STRESS_TEST==1            0
su_initialize.c:  0xA5C4   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5C1   STRESS_TEST==1            0
su_initialize.c:  0xA5Ce   STRESS_TEST==1            0
su_initialize.c:  0xA5F9   STRESS_TEST==1            0
su_initialize.c:  0xA5c9   CHECK_ACCUM_WSIA==1       0
su_initialize.c:  0xA5cA   CHECK_ACCUM_WSIA==1       0
su_initialize.c:  0xA5c0   TEST_RAM!=0               0
```

```
su_initialize.c:  0xA5c1   TEST_RAM!=0                0
su_initialize.c:  0xA5c2   TEST_RAM!=0                0
su_initialize.c:  0xA5c3   TEST_RAM!=0                0
su_initialize.c:  0xA5cc   TEST_RAM!=0                0
su_initialize.c:  0xA5cd   TEST_RAM!=0                0
su_initialize.c:  0xA5f0   STRESS_DATA_Q              undefined
su_initialize.c:  0xA5CC
su_locategs.c:    0xA5B1  DEBUG_LOCATEGS==1       0
whitedsp.c:       0xA5D2  DEBUG_RED_SAA==1       0
whitedsp.c:       0xA5D3  DEBUG_RED_SAA==1       0
whitedsp.c:       0xA5D4  DEBUG_RED_SAA==1       0
whitedsp.c:       0xA5D5  DEBUG_RED_SAA==1       0
whitedsp.c:       0xA5D6  DEBUG_RED_SAA==1       0
whitedsp.c:       0xA5D9  DO_FROG==1             0
```

The appearance of these alerts are very much dependent on the context DPU software and its debugging configuration. Consult the DPU software maintainer/developer for advice.

Instruction to DPU software developer/maintainer:

When turning on any compilation flag, be mindful of these extraneous alerts that have been turned on or any others that might have been implemented as a diagnostics tool. Inform the instrument operator/simulator not familiar with the internal working of the DPU to expect such extraneous alerts.

Appendix. Standard DPU Post Mortem (DPUPMM) procedure

In all cases of unscheduled DPU reset (Fred) and some severe exceptions alerts described above, it is necessary to capture the DPU state for post mortem failure analysis. The following is a sequence of DPU memory data extraction for this purpose.

```
=========================

prog_p0 = 0xe00000
size_p0 = 0xB000     # this number is determined by the DPU code compiler
prog_p0 = 0xe30000
size_p0 = 0x1000     # this number is determined by the DPU code compiler
prog_p0 = 0xe50000
size_p0 = 0x3400     # this number is determined by the DPU code compiler
prog_p7 = 0xe70000
size_p7 = 0x3400     # this number is determined by the DPU code compiler
prog_lut = 0xe7ff00
size_lut = 0x100     # this number does not change
```

```
proc_base = 0x0
size_proc = 0x8000   # this number does not change

dump_dpu_ram prog_p0, size_p0
dump_dpu_ram prog_p3, size_p3
dump_dpu_ram prog_p5, size_p5
dump_dpu_ram prog_p7, size_p7
dump_dpu_ram prog_lut, size_lut
dump_dpu_ram proc_base, proc_size
wait x seconds

where x  =
sum( size of p0, p3, p5, p7, lut, proc ) * 32 bits / expected TM rate

As of March 27, 1999. At rate of 8 kbits/sec, x = 240. Need more tuning.
```

11.12 Cheng's addendum

```
1.14.

#define         DA_BLUE1_E_ALERT         0xA594

This alert is not being used unless the bluedsp bad address (BADD)
compilation flag is turned on and code loaded with this debugging
procedure. Be aware of the code configuration. If BADD is set, consult the
DPU code maintainer. If the BADD is not set in the blue codes, this alert
could occur during start up if the DPU was reset from a strange state.
Under such circumstance, do nothing.

1.15.

#define         DA_BLUE1_S_ALERT         0xA595

This alert is not being used unless the bluedsp bad address (BADD)
compilation flag is turned on and code loaded with this debugging
procedure. Be aware of the code configuration. If BADD is set, consult the
DPU code maintainer. If the BADD is not set in the blue codes, this alert
could occur during start up if the DPU was reset from a strange state.
Under such circumstance, do nothing.

1.16.

#define         DA_BLUE2_E_ALERT         0xA596

This alert is not being used unless the bluedsp bad address (BADD)
compilation flag is turned on and code loaded with this debugging
procedure. Be aware of the code configuration. If BADD is set, consult the
DPU code maintainer. If the BADD is not set in the blue codes, this alert
could occur during start up if the DPU was reset from a strange state.
Under such circumstance, do nothing.

1.17.

#define         DA_BLUE2_S_ALERT         0xA597

This alert is not being used unless the bluedsp bad address (BADD)
compilation flag is turned on and code loaded with this debugging
procedure. Be aware of the code configuration. If BADD is set, consult the
DPU code maintainer. If the BADD is not set in the blue codes, this alert
could occur during start up if the DPU was reset from a strange state.
```

```
Under such circumstance, do nothing.

Special Note for alerts A594-A597:

It has been observed that a transient pair of alerts of DA_BLUE1_S_ALERT
were generated by the DPU without other severe detrimental effects such as
crashing or major change in operation configuration. It is believed that
this is an artifect of the particular message passing method between the
different DSPs in the DPU. The handling procedure of a transient pair of
error alerts from A594 through A597 is as follows.

a. When a pair is observed, mark the occurrence for post facto checkup and
continue normal operation.

b. Post facto checkup: The pair should be separated by one DPU
bus_cycle_count. Examine the status/error word to make sure that the first
word is a simple change (e.g. bit-flip) from the normal value and the
second occurrence carries the normal value. The normal values for
A594/A596 and A595/A597 are 0 and 0x300 respectively. If all of these
conditions are met, then this is a classical transient pair. Mark the
occurrence for long-term monitoring.

c. Long term monitor: If this transient alert pair occurs at a relatively
high regularity, this could indicate that the memory has become
susceptible for corruption. In this case, a simple change in the DPU code
will relocate the memory without loss of DPU functionality.

<<==

Hope this help clarify the issues.
```

12 General Debug Procedure

Do a low-level RBI DMA read of the ICU's memory (operand space).

These DMA reads of the ICU's memory can be done with the ICU even in a "crashed" state. The ICU's code does not need to be running correctly. It can be done at any time as long as the ICU is powered on.

12.1 ICU

Description	tc	tm	params	VxWorks
Spacecraft command: read the words in the OM ICU memory at addresses fc0$_{hex}$ to fdf$_{hex}$				`<debug`
Spacecraft command: read the words in the OM ICU memory at addresses a21$_{hex}$ to a25$_{hex}$				
Wait approx. 60 s				
Spacecraft command: read the words in the OM ICU memory at addresses fc0$_{hex}$ to fdf$_{hex}$				`<debug`

12.2 DPU

12.2.1 Dump Prog Area

Description	tc	tm	params	VxWorks
Reset DPU	**h7201**			`tc_reset_dsp`
Wait up to 10 s for alert		**92201**		Expect "Fred from the Dead"
Dump Bank 0 request	**h4120**		**h500=e00000**$_{hex}$ **h510=b000**$_{hex}$	`tc_dump_mem(0x20, 0xe00000, 0xb000)`
Dump Bank 3 request	**h4123**		**h500=e30000**$_{hex}$ **h510=1000**$_{hex}$	`tc_dump_mem(0x23, 0xe30000, 0x1000)`
Dump Bank 5 request	**h4125**		**h500=e50000**$_{hex}$ **h510=3400**$_{hex}$	`tc_dump_mem(0x25, 0xe50000, 0x3400)`
Dump Bank 7 request	**h4127**		**h500=e70000**$_{hex}$ **h510=3400**$_{hex}$	`tc_dump_mem(0x27, 0xe70000, 0x3400)`
Dump Bank 7 LUT request	**h4127**		**h500=e7ff00**$_{hex}$ **h510=100**$_{hex}$	`tc_dump_mem(0x27, 0xe7ff00, 0x100)`

12.2.2 Dump Proc area

Description	**tc**	**tm**	**params**	**VxWorks**
Dump Proc request	**h4114**		**h500=0 h510=8000**$_{hex}$	`tc_dump_mem(0x14, 0x0, 0x8000`

13 Appendix A: VxWorks Scripts

Here are the VxWorks scripts used for software testing in the UK and USA.

13.1 acquire

```
#############################################################################
#
# RCS $Id: acquire,v 1.5 1999/03/21 17:37:26 pjs1 Exp $
# RCS $Log: acquire,v $
# RCS Revision 1.5  1999/03/21 17:37:26  pjs1
# RCS Don't set exp_no and don't copy ref_stars.cfg file.
# RCS
# RCS Revision 1.4  1999/03/18 20:46:28  pjs1
# RCS Set threshold to 15.  Increased end delay.
# RCS
# RCS Revision 1.3  1999/03/15 15:26:41  pjs1
# RCS Decreased Jon delay (for simulator use only).
# RCS
# RCS Revision 1.2  1999/03/04 04:44:03  pjs1
# RCS Added load_tables.
# RCS
# RCS Revision 1.1  1999/03/04 02:28:45  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
#############################################################################


#################################################################
#
#   Set up some numbers first
#
#################################################################

date
frame_time = 15
bpe_bin = 0
filter = 1
#exp_no = 0xd40001
#copy "/home/xmmom/sim_data/ref_stars.cfg","ref_stars.cfg"

############################################

```

```
tc_det_acq_mode 0

# load det acq 0 configuration
< load_table_24
# Jon delay
taskDelay(10*60)

tc_fw_filter 1
tc_fw_move
taskDelay(20*60)

tc_set_bpe bpe_bin
tc_set_exp_no exp_no

tc_det_enbl_frame_tag 1
tc_det_event_threshold 15

# load reference stars
< load_table_25

# ref frame exp
tc_set_frame_time frame_time*1024

tc_send_command IC_INIT_EXP
taskDelay(3*60)

# change to science mode
#tc_mode 3
# start the detector integration
tc_det_start_int

tc_send_command IC_ACQUIRE_FLD
taskDelay(frame_time*60)
tc_det_stop_int
# should check this
taskDelay(60*60)
```

13.2 cgs

```
##############################################################################
#
# RCS $Id: cgs,v 1.4 1999/04/13 16:21:16 pjs1 Exp $
# RCS $Log: cgs,v $
# RCS Revision 1.4  1999/04/13 16:21:16  pjs1
# RCS Removed Jon delay.
# RCS
# RCS Revision 1.3  1999/03/18 20:46:50  pjs1
# RCS Added enable frame tags and threshold.
# RCS
# RCS Revision 1.2  1999/03/04 04:44:03  pjs1
# RCS Added load_tables.
# RCS
# RCS Revision 1.1  1999/03/04 02:28:46  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
##############################################################################

date
tc_fw_filter filter
tc_fw_move
taskDelay(20*60)
tc_det_acq_mode 0

# load acq 0 mode
< load_table_24
# Jon delay
taskDelay(20*60)

tc_set_exp_no exp_no
tc_send_command IC_INIT_EXP
taskDelay(3*60)

tc_det_enbl_frame_tag 1
tc_det_event_threshold 15

############
tc_set_frame_time frame_time*1024
# tc_prog_mem_wind "mmw.cfg"
< load_table_28
# tc_prog_sci_wind "scw.cfg"
< load_table_29
```

```
tc_set_exposure_time exposure_time
tc_set_bpe bpe_bin
tc_det_start_int
tc_send_command IC_CHOOSE_GS
taskDelay((frame_time+40+40)*60)
tc_det_stop_int
# Jon delay
#taskDelay(60*60)
```

13.3 exp

```
##############################################################################
#
# RCS $Id: exp,v 1.3 1999/04/13 16:21:31 pjs1 Exp $
# RCS $Log: exp,v $
# RCS Revision 1.3  1999/04/13 16:21:31  pjs1
# RCS Added tracking on.
# RCS
# RCS Revision 1.2  1999/03/18 20:47:05  pjs1
# RCS Changed threshold.
# RCS
# RCS Revision 1.1  1999/03/04 02:28:47  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
##############################################################################

date
tc_det_acq_mode 2
tc_set_bpe bpe_bin
#
# Put BPE into the right mode, really shouldn't be here
#
tc_det_enbl_frame_tag 1
tc_det_event_threshold 15
#
tc_det_start_int
# taskDelay 2*60
tc_send_command IC_TRACK_GS
taskDelay((frame_time*(exposure_time+1)+10)*60)
tc_set_trk 1
```

13.4 eng3

```
################################################################################
#
# RCS $Id: eng3,v 1.8 1999/04/13 16:14:52 pjs1 Exp $
# RCS $Log: eng3,v $
# RCS Revision 1.8  1999/04/13 16:14:52  pjs1
# RCS Changed detector windows.
# RCS
# RCS Revision 1.7  1999/03/21 17:38:48  pjs1
# RCS Changed archive name.  Changed exposure id.  Changed window size.  Added missing delay at end.
# RCS
# RCS Revision 1.6  1999/03/20 20:39:55  pjs1
# RCS Added open packet_archive. Moved LEDs. Changed exposure id.  Removed enable_verbose. Changed threshold.  Added a Dave.
# RCS
# RCS Revision 1.5  1999/03/17 21:19:54  pjs1
# RCS Added date, increased frame_time, added leds, added blocked, turned off frame
# RCS tags, increased delay, added Dave.
# RCS
# RCS Revision 1.4  1999/03/16 19:45:24  pjs1
# RCS Changed exposure ID, window coords, delays.
# RCS
# RCS Revision 1.3  1999/03/16 08:43:29  pjs1
# RCS Changed exposure id, acquisition mode, delay.
# RCS
# RCS Revision 1.2  1999/03/15 15:25:42  pjs1
# RCS Increased delay at end.
# RCS
# RCS Revision 1.1  1999/03/15 15:24:53  pjs1
# RCS New engineering mode scripts.
# RCS
#
################################################################################
date
open_packet_archive("/data/xmmom/archive/d40003")
frame_time=300
tc_mode 4

tc_init_dpu
taskDelay(60*25)
# expect Dave from the Grave

tc_fw_filter 0
tc_fw_move
taskDelay(20*60)
```

```
tc_det_flood_led_current 4
tc_set_frame_time frame_time*1024
tc_set_exp_no 0xd40003
tc_send_command IC_INIT_EXP
# tm_wait_for_alert DA_EOT_INIT_EXP
taskDelay(3*60)

det_reset_wind
# full size
#det_wind 64, 15, 256, 256
#inner part
det_wind 64+32, 15+32, 256-64, 256-64
tc_det_load_windows 1
taskDelay(60*60)

#turn on eng7 if necessary - probably not for post perigee procedure
#tc_enbl_verbose 4,1
tc_det_enbl_frame_tag 0
tc_det_event_threshold 15

tc_det_acq_mode 4

# start sending events from mic
tc_det_start_int

tc_enbl_eng 1, 3

# wait for DA_ENDOF_EXP DA_COMPLETE_EXP all data
taskDelay(frame_time*60)
# the following delay is for 4,1
#taskDelay(500*60)
taskDelay(380*60)

close_packet_archive
tc_det_flood_led_current 0

tc_mode 2
```

13.5 eng4

```
################################################################################
#
# RCS $Id: eng4,v 1.6 1999/03/21 17:39:12 pjs1 Exp $
# RCS $Log: eng4,v $
# RCS Revision 1.6  1999/03/21 17:39:12  pjs1
# RCS Changed archive name and exposure id.
# RCS
# RCS Revision 1.5  1999/03/20 20:43:47  pjs1
# RCS Changed frame_time. Added open_packet_archive. Added Dave.
# RCS
# RCS Revision 1.4  1999/03/17 21:20:45  pjs1
# RCS Added date, fixed delay, turned off frame tags.
# RCS
# RCS Revision 1.3  1999/03/16 19:50:37  pjs1
# RCS Changed threshold.
# RCS
# RCS Revision 1.2  1999/03/16 08:43:48  pjs1
# RCS Added mode change to idle.
# RCS
# RCS Revision 1.1  1999/03/15 16:47:26  pjs1
# RCS Engineering mode 4.
# RCS
#
################################################################################
date
open_packet_archive("/data/xmmom/archive/d40004")
frame_time=300
tc_mode 4

tc_init_dpu
taskDelay(60*25)
# expect Dave from the Grave

tc_set_frame_time frame_time*1024
tc_set_exp_no 0xd40004
tc_send_command IC_INIT_EXP
# tm_wait_for_alert DA_EOT_INIT_EXP
taskDelay(3*60)

det_reset_wind
det_wind 64, 15, 256, 256
tc_det_load_windows 1
taskDelay(60*60)
```

```
# Jon delay
taskDelay(20*60)

tc_det_enbl_frame_tag 0
tc_det_event_threshold 15
tc_det_acq_mode 3

# start sending events from mic
tc_det_start_int

tc_enbl_eng 1, 4

# wait for DA_ENDOF_EXP
taskDelay(frame_time*60)
#tm_wait_for_alert DA_COMPLETE_EXP
taskDelay(4800*60)
taskDelay(1000*60)

tc_mode 2
close_packet_archive
```

13.6 eng5

```
##############################################################################
#
# RCS $Id: eng5,v 1.6 1999/04/13 16:14:05 pjs1 Exp $
# RCS $Log: eng5,v $
# RCS Revision 1.6  1999/04/13 16:14:05  pjs1
# RCS Changed delays and exposure time.
# RCS
# RCS Revision 1.5  1999/03/21 17:39:31  pjs1
# RCS Changed archive name and exposure id.
# RCS
# RCS Revision 1.4  1999/03/20 20:44:52  pjs1
# RCS Lengthened frame_time. Added open_packet_archive. Moved LEDs. Changed threshold. Added Dave.
# RCS
# RCS Revision 1.3  1999/03/17 21:21:59  pjs1
# RCS Added date, increased frame_time, added blocked, disabled tags, added LEDs, added Dave.
# RCS
# RCS Revision 1.2  1999/03/16 19:45:24  pjs1
# RCS Changed exposure ID, window coords, delays.
# RCS
# RCS Revision 1.1  1999/03/15 15:24:53  pjs1
# RCS New engineering mode scripts.
# RCS
#
##############################################################################
date
open_packet_archive("/data/xmmom/archive/d40005")
# was 600
frame_time=4000
tc_mode 4

tc_init_dpu
taskDelay(60*25)
# expect Dave from the Grave

tc_fw_filter 0
tc_fw_move
taskDelay(20*60)
tc_det_flood_led_current 4
tc_set_frame_time frame_time*1024
tc_set_exp_no 0xd40005
tc_send_command IC_INIT_EXP
# tm_wait_for_alert DA_EOT_INIT_EXP
taskDelay(3*60)
```

```

det_reset_wind
det_wind 64, 15, 256, 256
tc_det_load_windows 1
taskDelay(60*60)

tc_det_enbl_frame_tag 0
tc_det_event_threshold 15
tc_det_acq_mode 3

# start sending events from mic
tc_det_start_int

tc_enbl_eng 1, 5

# wait for DA_ENDOF_EXP
taskDelay(frame_time*60)
taskDelay(130*60)
#tm_wait_for_alert DA_COMPLETE_EXP
taskDelay(20*60)
# wait for data
taskDelay(20*60)

close_packet_archive

tc_det_flood_led_current 0

tc_mode 2
```

13.7 eng6

```
################################################################################
#
# RCS $Id: eng6,v 1.6 1999/04/13 16:15:44 pjs1 Exp $
# RCS $Log: eng6,v $
# RCS Revision 1.6  1999/04/13 16:15:44  pjs1
# RCS Changed delay.
# RCS
# RCS Revision 1.5  1999/03/21 17:39:31  pjs1
# RCS Changed archive name and exposure id.
# RCS
# RCS Revision 1.4  1999/03/20 20:46:11  pjs1
# RCS Added open_packet_archive. Added Dave.
# RCS
# RCS Revision 1.3  1999/03/17 21:22:34  pjs1
# RCS Added date, added blocked, added LEDs, disabled frame tags, added Dave.
# RCS
# RCS Revision 1.2  1999/03/16 19:45:25  pjs1
# RCS Changed exposure ID, window coords, delays.
# RCS
# RCS Revision 1.1  1999/03/15 15:24:54  pjs1
# RCS New engineering mode scripts.
# RCS
#
################################################################################
date
open_packet_archive("/data/xmmom/archive/d40006")
frame_time=20
tc_mode 4

tc_init_dpu
taskDelay(60*25)
# expect Dave from the Grave

tc_fw_filter 0
tc_fw_move
taskDelay(20*60)
tc_det_flood_led_current 4

tc_set_frame_time frame_time*1024
tc_set_exp_no 0xd40006
tc_send_command IC_INIT_EXP
# tm_wait_for_alert DA_EOT_INIT_EXP
taskDelay(3*60)
```

```
det_reset_wind
det_wind 64, 15, 256, 256
tc_det_load_windows 1
taskDelay(60*60)

tc_det_enbl_frame_tag 0
tc_det_event_threshold 12
tc_det_acq_mode 6

# start sending events from mic
tc_det_start_int

tc_enbl_eng 1, 6

# wait (exposure time) for DA_ENDOF_EXP
taskDelay(frame_time*60)
#tm_wait_for_alert DA_COMPLETE_EXP
taskDelay(20*60)
# wait for end of data
taskDelay(20*60)

close_packet_archive

tc_det_flood_led_current 0

tc_mode 2
```

13.8 flatfield

```
###############################################################################
#
# RCS $Id: flatfield,v 1.3 1999/03/20 20:57:38 pjs1 Exp $
# RCS $Log: flatfield,v $
# RCS Revision 1.3  1999/03/20 20:57:38  pjs1
# RCS Added open_packet_archive. Changed exposure IDs. Altered LED. Put all exposures back in.
# RCS
# RCS Revision 1.2  1999/03/17 21:27:30  pjs1
# RCS Added date.
# RCS
# RCS Revision 1.1  1999/03/04 02:28:59  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
###############################################################################

date
open_packet_archive("/data/xmmom/archive/d40018")
#####################
# Common parameters for all 4 FF
#
frame_time = 25
exposure_time = 20
bpe_bin = 1
track = 0
filter = 0
led = 4
copy "/home/xmmom/sim_data/e40018/mmw.cfg","mmw.cfg"
#####################
tc_fw_filter 0
tc_fw_move
taskDelay(20*60)
tc_set_trk track
tc_set_bpe bpe_bin
tc_mode 2
tc_mode 4
tc_det_flood_led_current led
tc_mode 2
tc_mode 3
######################
# Specific parameters for FF
#
exp_no = 0xd40008
```

```
copy "/home/xmmom/sim_data/e40018/scw.cfg","scw.cfg"
######################
# could be <cgs_wo_fw_move to save some time
#
<cgs
<exp
######################
# Specific parameters for FF
exp_no = 0xd40009
copy "/home/xmmom/sim_data/e40019/scw.cfg","scw.cfg"
######################
# could be <cgs_wo_fw_move to save some time
#
<cgs
<exp
######################
# Specific parameters for FF
exp_no = 0xd4000a
copy "/home/xmmom/sim_data/e4001a/scw.cfg","scw.cfg"
######################
# could be <cgs_wo_fw_move to save some time
#
<cgs
<exp
######################
# Specific parameters for FF
exp_no = 0xd4000b
copy "/home/xmmom/sim_data/e4001b/scw.cfg","scw.cfg"
######################
# could be <cgs_wo_fw_move to save some time
#
<cgs
<exp
######################
tc_set_trk 1
tc_set_bpe 0
tc_mode 2
tc_mode 4
tc_det_flood_led_current 0
tc_mode 2
tc_mode 3

#########################################################################
#########################################################################
# vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
# remember to flush and close_packet_archive ############################
```

```
# ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
########################################################################
########################################################################
```

13.9 tc_clean_slate

```
##############################################################################
#
# RCS $Id: tc_clean_slate,v 1.2 1999/03/17 21:13:23 pjs1 Exp $
# RCS $Log: tc_clean_slate,v $
# RCS Revision 1.2  1999/03/17 21:13:23  pjs1
# RCS Added date and removed debug patch.
# RCS
# RCS Revision 1.1  1999/03/04 02:29:02  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
##############################################################################

date
#
#  Reset the ICU to the operational mode
#

rate 8

open_packet_archive "/data/xmmom/archive/d80ini"

set_dump_tm 0

####################################################
#
# This script brings up the DPU in a clean slate
#
####################################################

# clear all GSE semaphores

tm_clear_sem

#
#  First reset the DPU
#
tc_reset_dsp 0
taskDelay (60*12)

#
# Next reload the DPUOS
#
```

```
tc_load_dpuos
taskDelay (60*10)

# Turn on all color processors
tc_enbl_dsp red,1
taskDelay (60*10)

tc_enbl_dsp blue1,1
taskDelay (60*10)

tc_enbl_dsp blue2,1
taskDelay (60*10)

# Select blue2 as the klingon
tc_sel_klingon 1

# synchronize the clock bewteen ICU and DPU
tc_synch_clk_auto
taskDelay(2*60)

##################################################################
#
# Initialize DPU
#
##################################################################
tc_init_dpu
taskDelay(60*25)
# expect Dave from the Grave

# DA_TRK enabled by code - not selectable
#tc_report_tracking 1
# DP_TRH enabled already
#tc_enbl_verbose 0,1
# turn on DD_REF verbosity
tc_enbl_verbose 3,1

# remove this later. special patch to debug Linux/Sim
#tc_load_local_ram 0x5c60,0x30

close_packet_archive
```

13.10 dpu_post_mortem

```
##############################################################################
#
# RCS $Id: dpu_post_mortem,v 1.4 1999/03/20 20:15:55 pjs1 Exp $
# RCS $Log: dpu_post_mortem,v $
# RCS Revision 1.4  1999/03/20 20:15:55  pjs1
# RCS Removed unnecessary close_packet_archive.
# RCS
# RCS Revision 1.3  1999/03/17 21:27:20  pjs1
# RCS Added date.
# RCS
# RCS Revision 1.2  1999/03/04 05:24:04  pjs1
# RCS Added taskDelay after reset.
# RCS
# RCS Revision 1.1  1999/03/04 02:28:47  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
##############################################################################
#
#
date

open_packet_archive "/data/xmmom/archive/dpupmm"

speed = 3

tc_reset_dsp
taskDelay(10*60)
#### SU
base = 0xE00000
length = 0xB000
tc_dump_prog_ram base, length
#### Blue
base = 0xE30000
length = 0x1000
tc_dump_prog_ram base, length
#### Red
base = 0xE50000
length = 0x3400
tc_dump_prog_ram base, length
#### White DPUOS
base = 0xE70000
length = 0x3400
```

```
tc_dump_prog_ram base, length
#### LUT
base = 0xE7FF00
length =  0x0100
tc_dump_prog_ram base, length
#### PROC
tc_dump_ram 0x0000, 0x8000
###
taskDelay(60*180*speed)
close_packet_archive
```

13.11 load_icu

```
##############################################################################
#
# RCS $Id: load_icu,v 1.1 1999/03/04 02:29:00 pjs1 Exp $
# RCS $Log: load_icu,v $
# RCS Revision 1.1  1999/03/04 02:29:00  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
##############################################################################
rate 100
set_dump_tc 0
set_dump_tm 0
packet_load < /home/pjs1/icu/fm/oper/icu.hex
set_dump_tc 1
set_dump_tm 1
rate 8
```

13.12 load_dpu

```
##############################################################################
#
# RCS $Id: load_dpu,v 1.2 1999/03/20 20:58:12 pjs1 Exp $
# RCS $Log: load_dpu,v $
# RCS Revision 1.2  1999/03/20 20:58:12  pjs1
# RCS Removed unnecessary delay. Removed set_dump_tm.
# RCS
# RCS Revision 1.1  1999/03/04 02:28:59  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
##############################################################################
rate 100
set_dump_tc 0
set_dump_tm 0
tc_reset_dsp
taskDelay(60*10)
tc_packet_load 0 < /home/xmmom/dpu/development/eeprom/p0.cmd.0
tc_packet_load 5 < /home/xmmom/dpu/development/eeprom/p5.cmd.0
tc_packet_load 7 < /home/xmmom/dpu/development/eeprom/p7.cmd.0
tc_packet_load 3 < /home/xmmom/dpu/development/eeprom/blue_p3.cmd.0
set_dump_tc 0
set_dump_tm 0
#taskDelay(60*12)
tc_load_dpuos
taskDelay(60*6)
rate 8
```

13.13 load_table_24

```
##############################################################################
#
# RCS $Id: load_table_24,v 1.1 1999/03/04 04:44:04 pjs1 Exp $
# RCS $Log: load_table_24,v $
# RCS Revision 1.1  1999/03/04 04:44:04  pjs1
# RCS Added load_tables.
# RCS
#
##############################################################################
det_reset_wind
det_wind 64,15,128,128
det_wind 64,143,128,128
det_wind 192,15,128,128
det_wind 192,143,128,128
tc_det_load_windows 4
taskDelay(30*60)
```

13.14 load_table_25

```
###############################################################################
#
# RCS $Id: load_table_25,v 1.1 1999/03/04 04:44:04 pjs1 Exp $
# RCS $Log: load_table_25,v $
# RCS Revision 1.1  1999/03/04 04:44:04  pjs1
# RCS Added load_tables.
# RCS
#
###############################################################################
tc_prog_ref_stars "ref_stars.cfg"
```

13.15 load_table_28

```
##############################################################################
#
# RCS $Id: load_table_25,v 1.1 1999/03/04 04:44:04 pjs1 Exp $
# RCS $Log: load_table_25,v $
# RCS Revision 1.1  1999/03/04 04:44:04  pjs1
# RCS Added load_tables.
# RCS
#
##############################################################################
tc_prog_ref_stars "ref_stars.cfg"
```

13.16 load_table_29

```
##############################################################################
#
# RCS $Id: load_table_29,v 1.1 1999/03/04 04:44:05 pjs1 Exp $
# RCS $Log: load_table_29,v $
# RCS Revision 1.1  1999/03/04 04:44:05  pjs1
# RCS Added load_tables.
# RCS
#
##############################################################################
tc_prog_sci_wind "scw.cfg"
```

13.17 bounds

```
###############################################################################
#
# RCS $Id: bounds,v 1.4 1999/03/21 17:37:55 pjs1 Exp $
# RCS $Log: bounds,v $
# RCS Revision 1.4  1999/03/21 17:37:55  pjs1
# RCS Open archive.  Lengthened delay.
# RCS
# RCS Revision 1.3  1999/03/15 16:00:16  pjs1
# RCS Turned on verification.
# RCS
# RCS Revision 1.2  1999/03/15 15:27:30  pjs1
# RCS Altered delays.
# RCS
# RCS Revision 1.1  1999/03/04 02:28:45  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
###############################################################################
date
open_packet_archive("/data/xmmom/archive/bounds")
det_equal_bounds
tc_det_load_bounds 1
taskDelay(100*60)
# Jon delay
taskDelay(20*60)
close_packet_archive
```

13.18 todatum

```
##############################################################################
#
# RCS $Id: todatum,v 1.1 1999/03/17 21:12:41 pjs1 Exp $
# RCS $Log: todatum,v $
# RCS Revision 1.1  1999/03/17 21:12:41  pjs1
# RCS To datum script.
# RCS
# RCS
#
##############################################################################
date
tc_fw_to_datum
tc_fw_move
taskDelay(25*60)
```

13.19 hvonr

```
##############################################################################
#
# RCS $Id: hvonr,v 1.2 1999/03/21 17:40:02 pjs1 Exp $
# RCS $Log: hvonr,v $
# RCS Revision 1.2  1999/03/21 17:40:02  pjs1
# RCS Removed cathode to 2 V ramp.
# RCS
# RCS Revision 1.1  1999/03/17 21:08:17  pjs1
# RCS HV procedures.
# RCS
#
##############################################################################

date
tc_det_load_hv_ramp_params 3,100,25,0
tc_det_start_hv_ramp
taskDelay(10 * 60)

tc_det_load_hv_ramp_params 3,1500,25,0
tc_det_start_hv_ramp
taskDelay(90 * 60)

tc_det_load_hv_ramp_params 2,500,25,0
tc_det_start_hv_ramp
taskDelay(40 * 60)

# don't need
#tc_det_load_hv_ramp_params 1,2,25,0
#tc_det_start_hv_ramp
#taskDelay(8 * 60)

tc_det_load_hv_ramp_params 1,390,25,0
tc_det_start_hv_ramp
taskDelay(30 * 60)

tc_det_load_hv_ramp_params 2,550,25,0
tc_det_start_hv_ramp
taskDelay(10 * 60)

tc_det_load_hv_ramp_params 3,1850,25,0
tc_det_start_hv_ramp
taskDelay(30 * 60)
```

13.20 hvoff

```
#############################################################################
#
# RCS $Id: hvoff,v 1.1 1999/03/17 21:08:16 pjs1 Exp $
# RCS $Log: hvoff,v $
# RCS Revision 1.1  1999/03/17 21:08:16  pjs1
# RCS HV procedures.
# RCS
#
#############################################################################

date
tc_det_load_hv_ramp_params 1,0,25,0
tc_det_start_hv_ramp
taskDelay(30*60)

tc_det_load_hv_ramp_params 2,0,25,0
tc_det_start_hv_ramp
taskDelay(40*60)

tc_det_load_hv_ramp_params 3,0,25,0
tc_det_start_hv_ramp
taskDelay(100*60)
```

13.21 flush

```
################################################################################
#
# RCS $Id: flush,v 1.2 1999/03/10 22:13:19 mcrw Exp $
# RCS $Log: flush,v $
# RCS Revision 1.2  1999/03/10 22:13:19  mcrw
# RCS Added date.
# RCS
# RCS Revision 1.1  1999/03/04 02:28:59  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
################################################################################
date
tc_send_command IC_FLUSH_CMPRS
```

13.22 reset

```
################################################################################
#
# RCS $Id: reset,v 1.2 1999/03/17 21:27:31 pjs1 Exp $
# RCS $Log: reset,v $
# RCS Revision 1.2  1999/03/17 21:27:31  pjs1
# RCS Added date.
# RCS
# RCS Revision 1.1  1999/03/04 02:29:02  pjs1
# RCS The scripts after the Feb 1999 USA integration.
# RCS
#
################################################################################
date
reset
packet_msg 0
taskDelay(20*60)
packet_msg 1
```

14 Appendix B: NCR 94

```
From ho@lanl.gov Mon Feb  8 04:32:54 1999
Received: from msslaa.mssl.ucl.ac.uk (msslaa.mssl.ucl.ac.uk [128.40.70.177])
by mssls7.mssl.ucl.ac.uk (8.8.8+Sun/8.8.8) with ESMTP id EAA22064
for <pjs1@mssls7.mssl.ucl.ac.uk>; Mon, 8 Feb 1999 04:32:53 GMT
Received: from sst10a.lanl.gov (sst10a.lanl.gov [128.165.207.1])
by msslaa.mssl.ucl.ac.uk (8.9.1/8.9.1) with ESMTP id EAA22047
for <pjs1@mssls7.mssl.ucl.ac.uk>; Mon, 8 Feb 1999 04:32:51 GMT
Received: from sst10b.lanl.gov (ho@sst10b.lanl.gov [128.165.207.2])
by sst10a.lanl.gov (8.8.5/8.8.5) with ESMTP id VAA13317;
Sun, 7 Feb 1999 21:31:05 -0700 (MST)
Received: from ho@localhost with SMTP by sst10b.lanl.gov (8.8.5/2.4) id VAA23457; Sun, 7 Feb 1999 21:31:04 -0700 (MST)
Date: Sun, 7 Feb 1999 21:31:04 -0700 (MST)
From: Cheng Ho <ho@lanl.gov>
X-Sender: ho@sst10b
To: Tim Sasseen <tims@leo.physics.ucsb.edu>,
        Howard Huckle <heh@mssl.ucl.ac.uk>,
        Anthony Dibbens <apd@mssl.ucl.ac.uk>,
        Phil Smith <pjs1@mssls7.mssl.ucl.ac.uk>,
        "James R. M. Klarkowski" <jrklark@sandia.gov>
Subject: NCR94 resolution report
In-Reply-To: <Pine.SUN.3.96.990207204755.23186D-100000@sst10b>
Message-ID: <Pine.SUN.3.96.990207212504.23186F-100000@sst10b>
MIME-Version: 1.0
Content-Type: TEXT/PLAIN; charset=US-ASCII
Status: RO
X-Status: A
Content-Length: 2616
Lines: 63

Everyone, especially Howard and Jim, please review the following document.
We should submit this to ESA for approval and then we close NCR94.
----------------------------------------------------------
| Cheng Ho, Ph.D.                                        |
| Los Alamos National Lab. MS D436, Los Alamos, NM 87545 |
| Phone: (505)667-3904/667-6438, Fax: (505) 665-4414     |
| E-mail: ho@lanl.gov                                    |
----------------------------------------------------------

XMM-OM NCR94 Resolution Report
==============================

Problem statement
```

```
-----------------

The symptom we see is that the DPU will crash when we try to boot the red
or blue if the uploaded codes for red/blue contains an FEC Error. Current
suspiscion is that the error reporting mechanism between the color DSPs'
ROM code and the white DSP's DPUOS (Jim) codes. It is almost certain that
the problem resides on the ROM codes for the red and blue.

Possible resolutions
--------------------

1. The problematic codes resides on the space-qualified ROMs. The DPU
project does not have the resource to replace them.

2. The occurence of this error requires that the uploaded codes contain
an FEC error, which should be rare.

3. The problem is easily recognizable as rebooting the DPU will almost be
the first thing we do after we load the DPU codes. The validity of the
codes can be immediately checked. Codes can be immediately reloaded if
necessary.

These considerations lead to the following resolution approach: Develop
and implement a DPU code upload validation and recovery procedure, which
is described as follows.

XMM-OM DPU Code Upload and Validation Procedures
------------------------------------------------

1. The DPU codes are uploaded using the standard memory patch procedures.

2. Immediately after the DPU codes are loaded, load DPUOS and initialize
DPU. After initialization, the DPU red scrubbing will be activated and it
will protect against single bit error and report multiple bit error in the
program storage.

3. If the DPU resets during the post-code-load DPU initialization, then it
is likely that the uploaded codes contain FEC error(s). Immediately
perform a DPU code reload and reinitialization.

4. If the code load and initialization fail for two consecutive times,
perform the standard DPU post-mortem procedure to dump the DPU program
area for further diagnostics.

5. Whenever possible, leave the DPU in the DPUOS initialized state where
the red scrubbing is continuously exercised. This will help maintain the
```

`integrity of the DPU codes stored in the program RAM.`

15 Appendix C: NCR 106

In ICU Basic Mode, should the spacecraft issue a safing command, it will not conform to the standard format and the ICU will interpret the packet's CRC as the address to which it will jump leading to unpredictable results. Typically a reset or a watchdog kick.

The full text of the NCR follows:

"Mode transition telecommand (MFN 9001, GOTO SAFE) currently takes two parameters to specify safing level and start address of code to execute. The spacecraft CDMU will not supply these parameters, as it is based on an earlier version of the command."

16 Production Notes

This document was typeset by P. J. Smith on a AMD K6-2 350 MHz (o/c 400 MHz) PC running RedHat Linux 5.2 using LaTeX2ϵ (specifically the tetex-0.9.6 rpm).